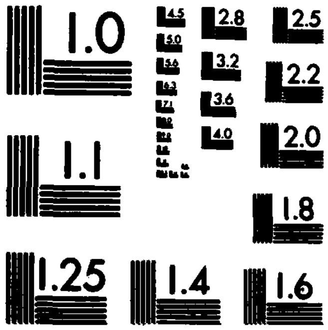
AD-A132 722 A USER'S GUIDE TO BISAM (BIVARIATE SAMPLE): THE BIVARIATE DATA MODELING P..(U) TEXAS A AND M UNIV COLLEGE STATION DEPT OF STATISTICS T J WOODFIELD 1/1

UNCLASSIFIED AUG 83 TR-A-24 ARO-20140.4-MA F/G 9/2 NL

END

FILMED



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

ARO 20140.4-MA

(12)

TEXAS A&M UNIVERSITY
COLLEGE STATION, TEXAS 77843-3143

STATISTICS
Phone 409 - 845-3141

A USER'S GUIDE TO BISAM: THE BIVARIATE DATA MODELING PROGRAM

AD-A133 711

by Terry J. Woodfield
Department of Statistics
Texas A&M University

Technical Report No. A-24
August 1983

Texas A&M Research Foundation
Project No. 4858

"Functional Statistical Data Analysis and Modeling"

Sponsored by the Army Research Office
Contract DAAG29-83-K-0051

Professor Emanuel Parzen, Principal Investigator

Approved for public release; distribution unlimited.

association along with some entropy measures that provide diagnostics for model selection and testing for independence. Graphical display of the estimated bivariate dependence density and density-quantile is the responsibility of the user, but a two step procedure utilizing the GCONTOUR and G3D procedures of SAS/GRAPH will be suggested in a later section. Different installations will have different software to provide three dimensional plots, hence the plotting option has been excluded from BISAM. One of the advantages of the modeling approach adopted by the BISAM program is the ability to obtain quickly and efficiently a set of function values for a dense grid of bivariate points promoting the examination of three dimensional representation of functions of interest.

A USER'S GUIDE TO BISAM: The Bivariate Data Modeling Program

I. Introduction

The BISAM (BIvariate SAMple) program is a companion program to the ONESAM (Parzen and Anderson, 1980) and TWOSAM (Prihoda, 1981) programs developed for nonparametric data modeling. The main purpose of BISAM is to perform bivariate data analysis using Fourier expansions and quantile techniques. The motivation and theory behind the data modeling approach incorporated by BISAM is detailed in Woodfield (1982), with foundations provided by Parzen (1979), Tartar and Kronmal (1970, 1976), and Kimeldorf and Sampson (1975).

BISAM is a FORTRAN program composed of a main program and 37 subprograms. Several of the subprograms come from ONESAM and the TIMESBOARD Time Series Subroutine Library (Newton, 1979). The main feature of BISAM is the ability to provide estimates of the bivariate density-quantile function of a set of bivariate data. A univariate analysis is also provided similar to that of ONESAM but with considerably less output. The univariate analysis is not intended to replace a ONESAM analysis if such a detailed analysis is required. The bivariate analysis includes display of various nonparametric measures of association along with some entropy measures that provide diagnostics for model selection and testing for independence. Graphical display of the estimated bivariate dependence density and density-quantile is the responsibility of the user, but a two step procedure utilizing the GCONTOUR and G3D procedures of SAS/GRAPH will be suggested in a later section. Different installations will have different software to provide

three dimensional plots, hence the plotting option has been excluded from BISAM. One of the advantages of the modeling approach adupted by the BISAM program is the ability to obtain quickly and efficiently a set of function values for a dense grid of bivariate points promoting the examination of three dimensional representation of functions of interest.

This document will explain the algorithms employed in generating output and the steps necessary to perform an analysis of a set of bivariate data. The options available and the user input required to execute the BISAM program will also be described in detail. The user should be thoroughly acquainted with the input options before attempting to execute the BISAM program. Incorrectly specified option values may cause errors. Furthermore, BISAM is a very long and complex program that will be fairly expensive to use, and hence improper runs should be avoided.

2. Univariate Analyses

The analysis of each of the paired variables separately is essential to fully understand the nature of the bivariate data set. In particular, the estimation of univariate density-quantile functions should be performed in an optimal manner to insure that the estimated bivariate density-quantile function is appropriate. A univariate analysis will provide diagnostics to help one understand the nature of the univariate density-quantile function and thus guide one in the formulation of the bivariate density-quantile.

BISAM provides output for each variable that is a subset of what may be provided by ONESAM. The univariate analysis is done in two stages, however. In the first stage, descriptive statistics, an Informative Quantile plot, and a plot of $\tilde{D}(u)$ are provided in a goodness-of-fit exploration analysis to guide one in determining the underlying marginal distributions.

In the second stage, a univariate density-quantile function is estimated by the autoregressive method for the null case specified and is then used to form the bivariate density-quantile function as described in section 4. If $\tilde{D}(u)$ in stage one does not approximate a uniform distribution function for the null case specified, then one should specify another null value, or the bivariate density-quantile function estimated will be unreliable.

For a complete description of some of the above concepts, see Parzen (1979).

3. Bivariate Analysis

A bivariate analysis of a set of data includes computation of various nonparametric correlation coefficients, model selection diagnostics, and estimation of the bivariate dependence density and density quantile functions for a dense grid of points. Scatter plots are produced for the original data and the rank transformed data. Obtaining "publication quality" plots depends on the plotting hardware and software available on the system used.

Five correlation coefficients are produced along with various entropy measures of association. We will assume that the input data set is denoted by $(X_1, Y_1),\ldots,(X_n, Y_n)$. Pearson's product moment correlation coefficient is defined by

$$r = \sum_{i=1}^{n} (X_i - \bar{X})(Y_i-\bar{Y}) \Big/ \sqrt{\sum_{i=1}^{n} (X_i-\bar{X})^2 \sum_{i=1}^{n} (Y_i-\bar{Y})^2} ,$$

and is labeled "PEARSON" in the BISAM output. Spearman's rank correlation coefficient is given by

$$\rho_n = 1 - 6 \sum_{i=1}^{n} (Q_i - R_i)^2 / [n(n^2 - 1)],$$

where Q_i = rank (X_i) and R_i = rank (Y_i) and is labeled "SPEARMAN" in the BISAM output. Three correlation coefficients related to the concepts of concordance and discordance are also computed, two of which perform corrections for tied observations. Kendall's Tau is defined by

$$\tau_A = (N_C - N_D)/[n(n-1)/2] \quad ,$$

where N_C is the number of concordant pairs and N_D is the number of discordant pairs, and is labeled "KENDALL A" in the output. Kendall's TAU correcting for ties is given by

$$\tau_B = (N_C - N_D)/ \sqrt{(N_C + N_D + T_X)(N_C + N_D + T_Y)},$$

where T_X is the number of tied observations in the X variable but not in the Y variable, and T_Y is the number of tied observations in the Y variable but not in the X variable. This measure is labeled "KENDALL B" in the output. Somer's d is defined by

$$d = (N_C - N_D) \;/\; (N_C + N_D + T_Y) \quad ,$$

and is seen to be similar to Kendall's Tau-B. This measure of association is labeled "SOMER'S D" in the output.

The ranking procedure employed assigns average ranks for tied observations. Other methods for assigning ranks to tied observations are often employed but are not attempted by BISAM. The presence of a large percentage of tied

observations will weaken the results obtained since underlying continuous distributions are assumed. One should avoid such situations if possible.

Two methods of bivariate density estimation are performed by BISAM, the nearest neighbor technique and the orthogonal expansion technique of Woodfield (1982). However, estimates obtained from the nearest neighbor technique are not displayed, but instead are used as input values for the orthogonal expansion technique. Since the density estimation methods employed are fundamental to the bivariate analysis performed, the next section will discuss these techniques with considerable attention paid to the theory and implementation of such methods. Since the entropy measures of association and model selection diagnostics are so closely related to the density estimation technique employed, the discussion of these quantities will also be withheld until the next section.

4. Nonparametric Density Estimation

Nonparametric density estimation is performed three different ways by BISAM for different purposes. The k-nearest neighbor technique is employed first for k=6 to provide an estimate of the dependence density as input to an orthogonal expansion density estimation technique and to produce a raw estimate of the entropy associated with the joint probability density function (p.d.f.) and the marginal p.d.f.'s. The orthogonal expansion method of Woodfield (1982) is then employed to obtain smoothed estimates of the dependence density for various degrees of smoothing. Finally, autoregressive estimates of the marginal p.d.f.'s are obtained and used with the estimated dependence density to produce an estimate of the bivariate density-quantile function. We now elaborate on the details of these procedures.

Let $(X_1, Y_1),\ldots,(X_n, Y_n)$ be a bivariate random sample form the random vector (X,Y) with joint cumulative d.f. $F_{X,Y}$, marginals F_X, F_Y, joint p.d.f. $f_{X,Y}$, marginal p.d.f.'s f_X, f_Y, and quantile functions Q_X, Q_Y. Define the dependence distribution function $D(u_1, u_2)$ by

$$D(u_1, u_2) = F_{X,Y}(Q_X(u_1), Q_Y(u_2)), \quad 0 \le u_1, u_2 \le 1 \quad ,$$

and the dependence density function $d(u_1,u_2)$ by

$$d(u_1,u_2) = \frac{\partial^2}{\partial u_1 \partial u_2} D(u_1,u_2) = \frac{f_{X,Y}(Q_X(u_1)Q_Y(u_2))}{f_X Q_X(u_1) f_Y Q_Y(u_2)} \quad .$$

Define the information between two densities f and g by

$$I(f;g) = \int_{-\infty}^{\infty} \{\log \frac{f(x)}{g(x)}\} f(x) \quad dx$$

and the entropy of a density f by

$$H(f) = \int_{-\infty}^{\infty} -\{\log f(x)\} f(x) \, dx.$$

The information inequality states that for two densities f and g,

$$I(f;g) \geq 0.$$

Furthermore, $I(f;g) = 0$ iff $f=g$ a.e.

It is easy to see that

$$I(f_{X,Y};\ f_X f_Y) = -H(d) \quad ,$$

which justifies naming $d(u_1,u_2)$ the dependence density, since by virtue of the information inequality $d(u_1,u_2)$ is indirectly related to a measure of dependence between X and Y. This fact is exploited in obtaining diagnostics for model selection and tests of independence.

Let $\{\phi_k(u)\}_{k=-\infty}^{\infty}$ be a complete orthonormal system of functions in $L^2(0,1)$. Then the system $\{\phi_{jk}(u_1,u_2)\}_{j,k=-\infty}^{\infty}$ defined by

$$\phi_{jk}(u_1,u_2) = \phi_j(u_1)\phi_k(u_2) \quad , \quad 0\le u_1,u_2\le 1, \quad \text{all } j,k,$$

is a complete orthonormal system in the space of bivariate square integrable functions on the unit square. If $\log d(u_1,u_2)$ is square integrable, then

$$\log d(u_1,u_2) = \sum_{j,k=-\infty}^{\infty} \theta_{jk}\phi_j(u_1)\phi_k(u_2) - \Psi(\underset{\sim}{\theta})$$

in the sense of L^2 norm where $\{\theta_{jk}\}_{j,k=-\infty}^{\infty}$ are the Fourier coefficients defined by

$$\theta_{jk} = \int_0^1 \int_0^1 \phi_j(u_1)\phi_k(u_2) \log d(u_1,u_2)\, du_1 du_2, \; j,k=-\infty, \ldots \infty \quad .$$

The term $\Psi(\underline{\theta})$ is included to insure that $d(u_1,u_2)$ integrates to one. For the truncated m-th order model given by

$$\log d_m(u_1,u_2) = \sum_{j,k=-m}^{m} \theta_{jk}\phi_j(u_1)\phi_k(u_2) - \Psi_m(\underline{\theta}) ,$$

it follows that $\log d_m(u_1,u_2) \to \log d(u_1,u_2)$ as $m \to \infty$, and hence one calls $d_m(u_1,u_2)$ the m-th order approximation of $d(u_1,u_2)$. One thus seeks an estimator of $d_m(u_1,u_2)$ based on a random sample of bivariate data.

Let

$$u_{1i} = R_i/(n+1), \qquad u_{2i} = Q_i/(n+1)$$

where R_i and Q_i are the ranks of X_i and Y_i as defined before. Thus $(u_{11}, u_{21}),\ldots,(u_{1n},u_{2n})$ approximates a random sample distributed uniformly on the unit square under an assumption of independence. Dependence of X and Y suggests alternate uniform distributions that may include a variety of bivariate density shapes from distributions having uniform (0,1) marginals. Next, form $\tilde{d}(u_1,u_2)$ evaluated at these uniform sample points based on the k-nearest neighbor techniques. BISAM uses k=6. Form $\log \tilde{d}(u_1,u_2)$ and use this as the dependent variable in a least squares regression routine with independent variables $\phi_j(u_1)\phi_k(u_2)$, $j,k=-m,\ldots,m$. Essentially, both $\log \tilde{d}(u_1,u_2)$ and $\phi_j(u_1)\phi_k(u_2)$ represent bivariate functions imbedded into univariate representations for a multiple regression analysis. The design matrix X is an $n \times m^2$ matrix consisting of the m^2 orthogonal combinations evaluated at the n uniform data points. The routine performed by BISAM forms the $(2m+1)^2 \times (2m+1)^2$ correlation matrix* and then uses a SWEEP operator

*The (0,0) term is incorporated into the integration factor and is not included, i.e., the no intercept model is employed, hence, the correlation matrix is not $[(2m+1)^2+1] \times [(2m+1)^2+1]$.

on this matrix to obtain least squares estimates of the Fourier coefficients. The resulting estimate is given by

$$\log \hat{d}(u_1,u_2) = \sum_{j,k=-m}^{m} \hat{\theta}_{jk}\phi_j(u_1)\phi_k(u_2)$$

where $\{\hat{\theta}_{j,k}\}_{j,k=-m}^{m}$ is the collection of least squares estimates of the expansion parameters. The integration factor $\hat{\Psi}_m(\hat{\underset{\sim}{\theta}})$ is then derived to insure that $d(u_1,u_2)$ numerically integrates to one. This is accomplished for $\hat{d}(u_1,u_2)$ equally spaced in the u_1 and u_2 directions within the unit square.

Three models are obtained for m=1,2,3 which translates into 8 variable, 24 variable, and 48 variable regression models, i.e., the terms with subscripts (i,k), i,k=-m,...,m for m=1,2,3 are included. These three models produce three density estimates evaluated at the 40 x 40 grid of (u_1,u_2) coordinates mentioned above. Using raw Riemann sums as numerical integrals, one then obtains estimates of the entropy H(d) mentioned earlier. A fourth estimate of the entropy of the dependence density is provided by the nearest neighbor estimate using the formula

$$H(\tilde{d}) = \int_0^1 \int_0^1 \log \tilde{d}(u_1,u_2)d\, D_n(u_1,u_2)$$

$$= \frac{1}{n} \sum_{j=1}^{n} \log \tilde{d}\left(\frac{R_j}{n+1}, \frac{Q_j}{n+1}\right) \quad ,$$

where $D_n(u_1,u_2)$ is the empirical c.d.f. of (u_{1i}, u_{2i}), i=1,...,n.

As a rule-of-thumb for model selection, $H(\hat{d})$ is compared with $H(\hat{d}_8)$, $H(\hat{d}_{24})$, and $H(\hat{d}_{48})$. An information criterion modeled after Akaike's Information Criterion (AIC) familiar to time series analysts is used to select the "best" model. One forms

$$AIC(k) = H(\hat{d}_k) - H(\tilde{d}) - 2k/n \quad ;$$

and then selects model k for the k value at which AIC(k) achieves its positive minimum. An alternative approach chooses k for which $|AIC(k)|$ is minimum. Model selection criterion are still under investigation with promising results having been achieved for the criterion employed by BISAM.

Graphical displays of $\hat{d}_k$ suffer from the unruly behavior of $\hat{d}_k$ near the boundary of the unit square, and hence one may prefer to display values of the bivariate density-quantile function given by

$$f\hat{Q}(u_1,u_2) = \hat{d}_k(u_1,u_2)\ f\hat{Q}_X(u_1)\ f\hat{Q}_Y(u_2)$$

where $f\hat{Q}_X$ and $f\hat{Q}_Y$ are the autoregressive estimates of the univariate density-quantile functions. Figure 1 contains a three dimensional plot of the bivariate density-quantile function of order 8 for a set of simulated bivariate normal random variables with correlation coefficient $\rho = 0.9$. Figure 2 contains the theoretical bivariate density-quantile function for the simulated data. The plots were produced using SAS/GRAPH. Appendix B outlines a two step FORTRAN-SAS procedure that allows one to produce three dimensional or contour plots of bivariate functions of interest.

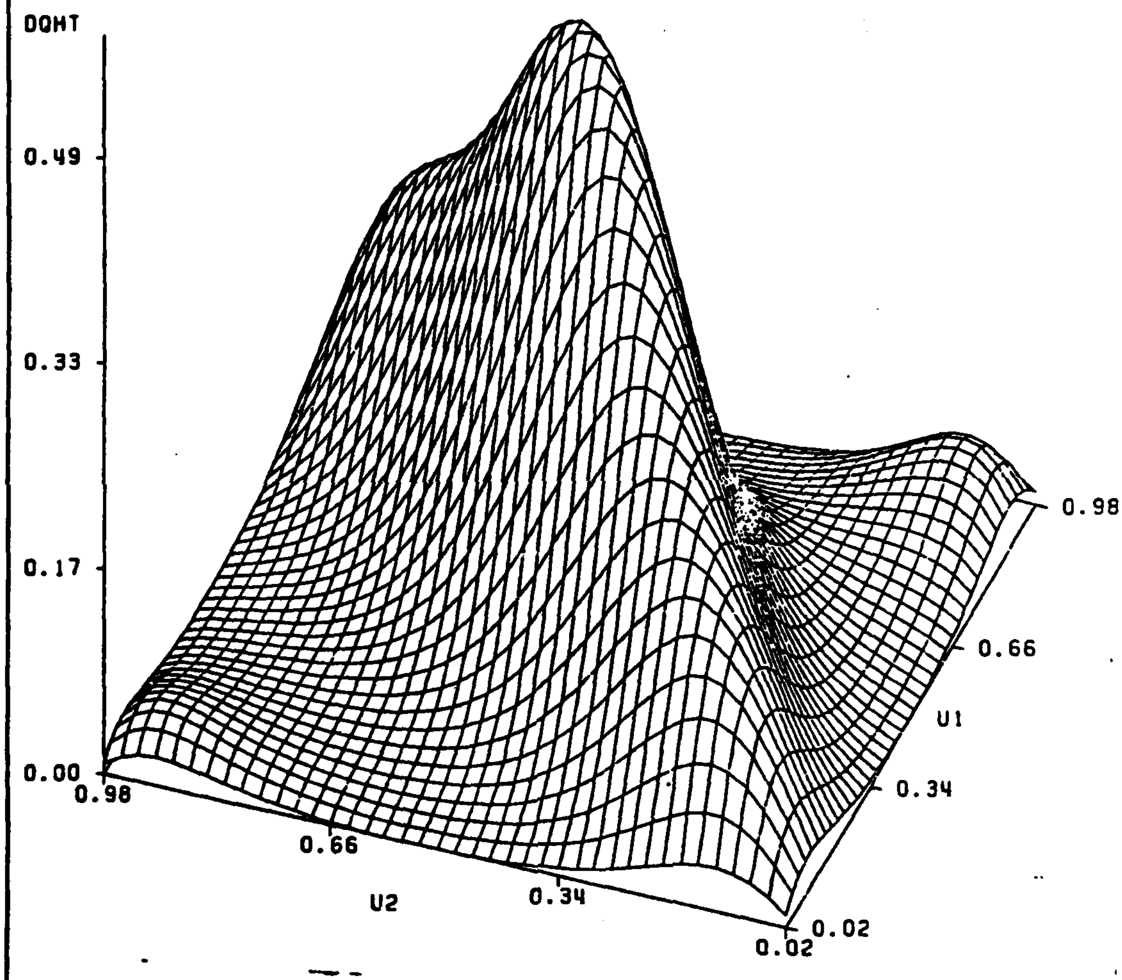


Figure 1. Three Dimensional Plot of Density-Quantile for Normal Data with Rho=0.9, Order=8

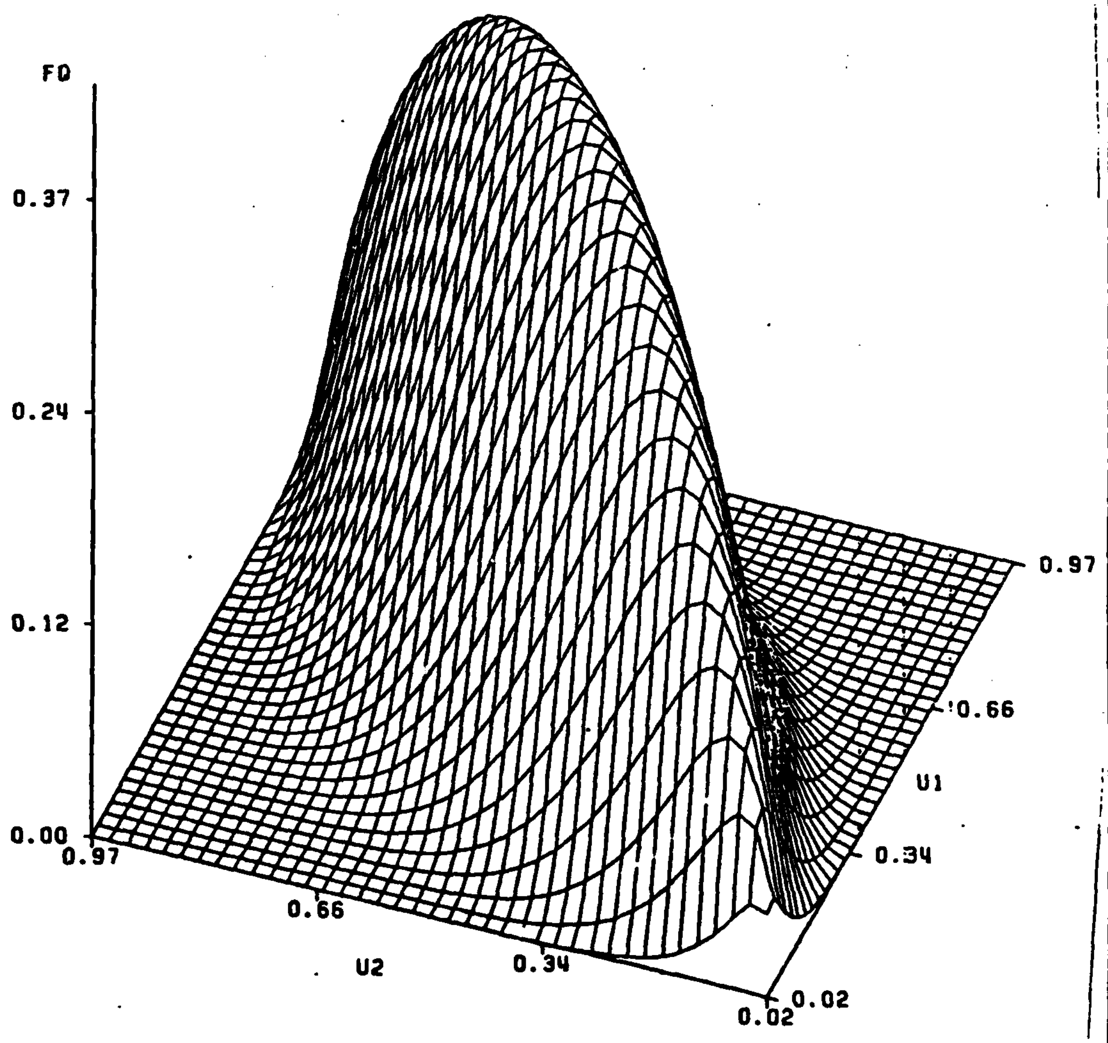


Figure 2. Bivariate Standard Normal Density-Quantile for Rho=0.9

5. Using BISAM to Analyze Time Series Data

Although primarily intended to analyze bivariate data in which the data values are uncorrelated, BISAM can also be used to analyze time series data. For a univariate data set one may use option IUNIV to do an analysis of a bivariate data set consisting of the univariate data set paired with lagged values of the data set.

Consider a Guassian time series $X(t)$ with autocorrelation function $\rho(v)$ given by

$$\rho(v) = \mathrm{Corr}[X(t),\ X(t+v)], \quad v=0,\ \pm 1,\ \ldots \quad .$$

Define the series $Y(t)$ to be a lagged version of $X(t)$, i.e., $Y(t) = X(t+v)$. Then

$$\rho_{X,Y} = \mathrm{Corr}[X,Y] = \mathrm{Corr}[X(t),\ X(t+v)] = \rho(v).$$

For a bivariate normal random vector (X,Y) with correlation coefficient $\rho_{X,Y}$, it can be shown that

$$I(f_{X,Y}\ ;\ f_X f_Y) = -.05 \log\ (1-\rho^2_{X,Y}) \quad .$$

Consequently, the information between X and lagged values of X is

$$I(v) = -0.5 \log\ (1-\rho^2(v)) \quad .$$

In the non-normal case, this equation does not hold so that a nonparametric

estimator may be desired to indicate the relationship between X and lagged values of X.

When one specifies IUNIV = 1, values for KLAG1 and KLAG2 must also be specified. Then a bisam analysis is performed on (X(t), X(t+v)) for integer values v satisfying KLAG1 $\leq$ v $\leq$ KLAG2.

6. Input Options

The following options are input on the first data card in 11I5 format. They are input in the order listed, and if NTAPE = 5, the data set follows this card in the indicated format listed at the end of this section.

NTAPE - number referring to DD statement describing the input data set.

IDQX - null distribution for autoregressive smoothing for X input variable.

IDQY - same as IDQX except for Y input variable.

MORD - maximum autoregressive order to be used for univariate autoregressive density estimation ($\leq$ 6).

IPLT1 - 0 for no scatter plots.
1 for scatter plot of data.
2 for scatter plot of rank transformed data.
3 for both scatter plots.

IPLT2 - 0 for no univariate density plots.
1 for best order AR univariate density plots.

IDST - 0 for no univariate descriptive statistic.
1 for descriptive statistics for both X and Y.

KDEL - maximum number of extreme points to exclude from bivariate analysis. An extreme point is located based on the distance of the X or Y coordinate from its median. If KDEL=2 and both the extreme X and Y are paired together, then only that point will be omitted. If KDEL=3, two points with an extreme X value and one point with an extreme Y value will be omitted, i.e., the X-direction received precedence over the Y-direction for odd values of KDEL. This method of deleting outliers works well for nearly linear relationships between X and Y. Its use is questionable for other cases.

IOUTD - 0 if function values are not to be saved.

1 if function values for the three fitted models are to be written to tapes 1, 2, and 3 respectively.

If IOUTD=1, the JCL must contain three DD cards for FTØ1FØØ1, FTØ2FØØ1, and FTØ3FØØ1 defining permanent dish data files where the function values are to be written. Typically, 10 tracks of storage must be alloted for each file.

IREG - 0 if no quantile regression performed.

1 if nonparametric estimates of $rQ(u) = E[Y|X = Q(u)]$ and $r(x) = E[Y|X=x]$ are desired.

IUNIV - 0 if input data set is bivariate data.

1 if input data set is a univariate time series as described in section 5.

If IUNIV=1, a second data card is required containing the values KLAG1 and KLAG2 in 2I5 format as described in section 5.

If IDQX and/or IDQY specify the Pareto to Weibull distributions, a separate data card must be inserted containing estimates of the parametrs in those distributions. For the Weibull, the location estimate is given first, with the input format specified as 2F1Ø.Ø.

The input data set consists of two univariate data sets containing the same number of values. It is always assumed that the X data set is read in first. The values (X,Y) are then paired by the order they are read in. A univariate data set consists of a title card read in 20A4 format, a description card containing the number of data points and the format of the input data read in (I5, 4X, 5A4) format, and the data cards containing the data coded in the format indicated on card 3. If the number of data points indicated for X and Y do not match, theprogram will terminate.

Example of univariate data set:

```
YEARLY SNOWFALL IN BUFFALO, 1910-1972
   63    (6F10.2)
      71.2      69.5      47.8      58.4      29.9      42.5
      etc.
```

The code for BISAM and sample output are contained in the Appendices.

7. Sample Output form BISAM

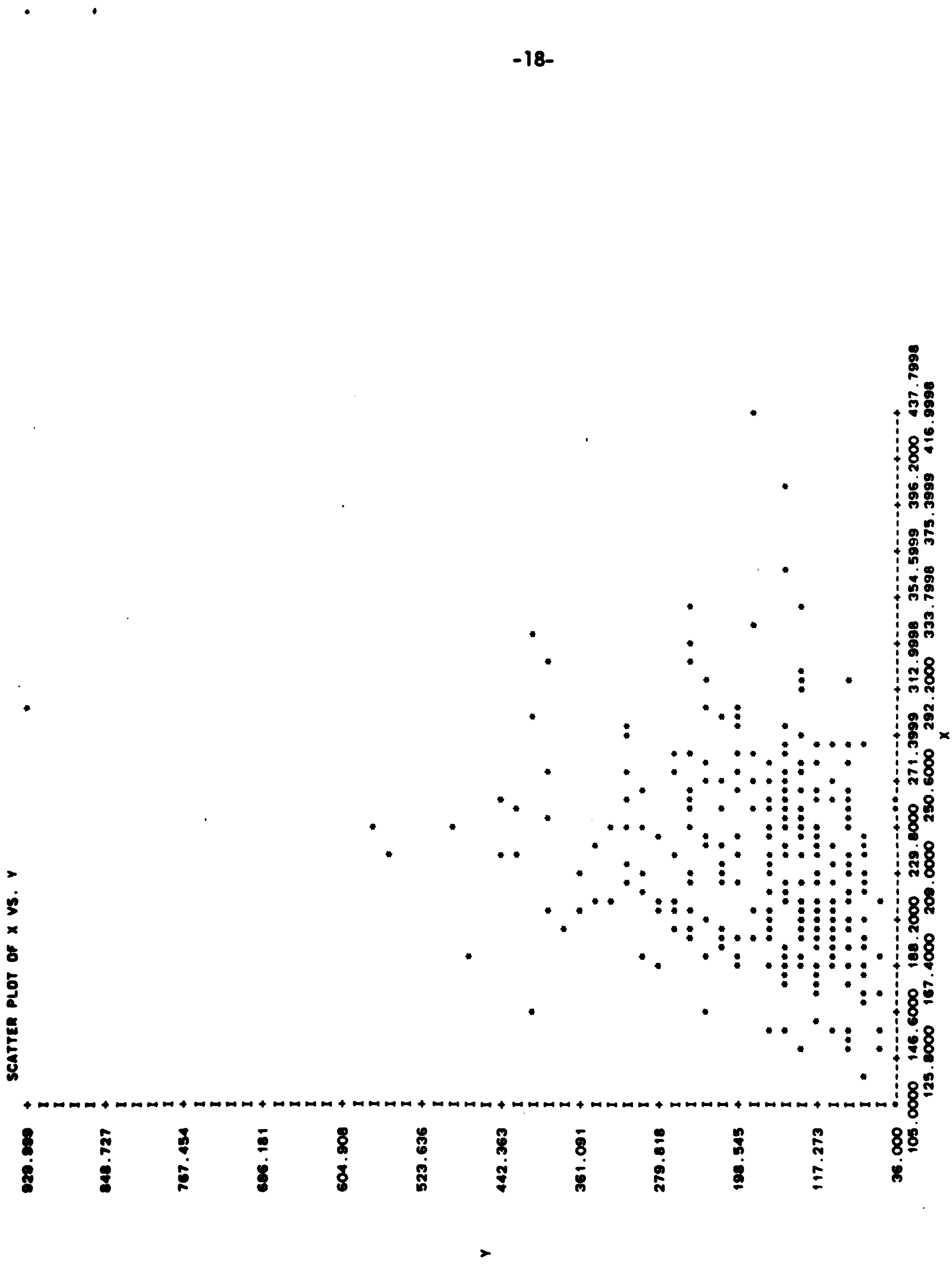
Following is a listing of the output from BISAM for a typical run with the input options clearly labeled. The JCL for executing BISAM at Texas A&M University is given in Appendix A.

```
*****************************************************************************
*                                                                           *
*   BISAM - BIVARIATE DATA ANALYSIS USING FOURIER EXPANSIONS                *
*           AND QUANTILE TECHNIQUES                                         *
*                                                                           *
*   PROGRAMMER: TERRY J. WOODFIELD                                          *
*                                                                           *
*****************************************************************************


 PLASMA CHOLESTEROL - DISEASE IN AT LEAST 1 OF 3 CORONARY ARTERIES
 PLASMA TRIGLYCERIDES - DISEASE IN AT LEAST 1 OF 3 CORONARY ARTERIES

SAMPLE SIZE =  320

OPTIONS FOR THIS ANALYSIS:
         NTAPE = 13          IDOX =    1          IDOY =  1
         MORD =  2           IPLT1 =   3          IPLT2 =  1
         IDST =  1           KDEL =    1          IOUTD =   0
         IREG =   0          IUNIV =    0
```


SCATTER PLOT OF X VS. Y
Y
929.999
848.727
767.454
686.181
604.908
523.636
442.363
361.091
279.818
198.545
117.273
36.000
105.0000
125.8000
146.6000
167.4000
188.2000
209.0000
229.8000
250.6000
271.3999
292.2000
312.9998
333.7998
354.5999
375.3999
396.2000
416.9998
437.7998
X

FULLY NON-PARAMETRIC ANALYSIS

PLASMA CHOLESTEROL - DISEASE IN AT LEAST 1 OF 3 CORONARY ARTERIES
ORIGINAL DATA - X

ORDER STATISTICS IN QUARTERS

SEQUENCE WITHIN QUARTILE	FIRST QUARTER	SECOND QUARTER	THIRD QUARTER	FOURTH QUARTER
1	105.0000	185.0000	213.0000	242.0000
2	119.0000	185.0000	213.0000	243.0000
3	131.0000	186.0000	214.0000	243.0000
4	131.0000	187.0000	215.0000	243.0000
5	131.0000	187.0000	215.0000	244.0000
6	138.0000	188.0000	216.0000	244.0000
7	139.0000	188.0000	216.0000	245.0000
8	139.0000	189.0000	217.0000	245.0000
9	140.0000	189.0000	217.0000	245.0000
10	140.0000	189.0000	218.0000	245.0000
11	140.0000	190.0000	218.0000	245.0000
12	142.0000	191.0000	218.0000	246.0000
13	144.0000	191.0000	218.0000	247.0000
14	149.0000	191.0000	219.0000	247.0000
15	150.0000	191.0000	219.0000	248.0000
16	151.0000	191.0000	219.0000	248.0000
17	157.0000	191.0000	220.0000	249.0000
18	159.0000	192.0000	220.0000	249.0000
19	159.0000	193.0000	220.0000	249.0000
20	160.0000	193.0000	221.0000	250.0000
21	162.0000	193.0000	221.0000	250.0000
22	163.0000	193.0000	221.0000	250.0000
23	164.0000	194.0000	221.0000	251.0000
24	164.0000	194.0000	221.0000	251.0000
25	165.0000	194.0000	222.0000	251.0000
26	165.0000	194.0000	222.0000	252.0000
27	165.0000	194.0000	222.0000	253.0000
28	167.0000	194.0000	222.0000	254.0000
29	168.0000	194.0000	223.0000	254.0000
30	168.0000	195.0000	223.0000	255.0000
31	168.0000	195.0000	224.0000	257.0000
32	168.0000	196.0000	225.0000	258.0000
33	169.0000	196.0000	226.0000	258.0000
34	169.0000	196.0000	227.0000	258.0000
35	170.0000	197.0000	227.0000	258.0000
36	171.0000	197.0000	227.0000	259.0000
37	171.0000	197.0000	228.0000	260.0000
38	171.0000	197.0000	228.0000	260.0000
39	171.0000	197.0000	228.0000	260.0000
40	171.0000	197.0000	229.0000	261.0000
41	171.0000	198.0000	229.0000	262.0000
42	171.0000	198.0000	230.0000	262.0000
43	172.0000	198.0000	230.0000	263.0000

44	172.0000	198.0000	230.0000	264.0000
45	172.0000	198.0000	230.0000	264.0000
46	172.0000	199.0000	230.0000	265.0000
47	173.0000	200.0000	230.0000	266.0000
48	173.0000	200.0000	230.0000	267.0000
49	174.0000	200.0000	231.0000	267.0000
50	175.0000	201.0000	231.0000	268.0000
51	175.0000	201.0000	231.0000	269.0000
52	175.0000	203.0000	232.0000	270.0000
53	175.0000	204.0000	232.0000	271.0000
54	176.0000	204.0000	232.0000	271.0000
55	176.0000	206.0000	232.0000	273.0000
56	177.0000	206.0000	232.0000	274.0000
57	178.0000	206.0000	233.0000	276.0000
58	178.0000	206.0000	233.0000	278.0000
59	178.0000	207.0000	233.0000	279.0000
60	178.0000	207.0000	233.0000	280.0000
61	178.0000	208.0000	233.0000	283.0000
62	179.0000	208.0000	234.0000	283.0000
63	179.0000	208.0000	235.0000	284.0000
64	180.0000	208.0000	236.0000	285.0000
65	180.0000	208.0000	236.0000	287.0000
66	180.0000	208.0000	237.0000	294.0000
67	181.0000	208.0000	237.0000	297.0000
68	181.0000	209.0000	238.0000	298.0000
69	184.0000	209.0000	239.0000	299.0000
70	184.0000	209.0000	239.0000	304.0000
71	184.0000	210.0000	239.0000	306.0000
72	184.0000	210.0000	239.0000	308.0000
73	184.0000	210.0000	239.0000	313.0000
74	185.0000	211.0000	240.0000	319.0000
75	185.0000	211.0000	240.0000	323.0000
76	185.0000	211.0000	240.0000	331.0000
77	185.0000	211.0000	242.0000	332.0000
78	185.0000	211.0000	242.0000	348.0000
79	185.0000	212.0000	242.0000	386.0000
80	185.0000	212.0000	242.0000	417.0000
SUM	13318.0000	15923.0000	18226.0000	21713.0000
SUM OF SQUARES	2240472.00	3174275.00	4157714.00	5973541.00

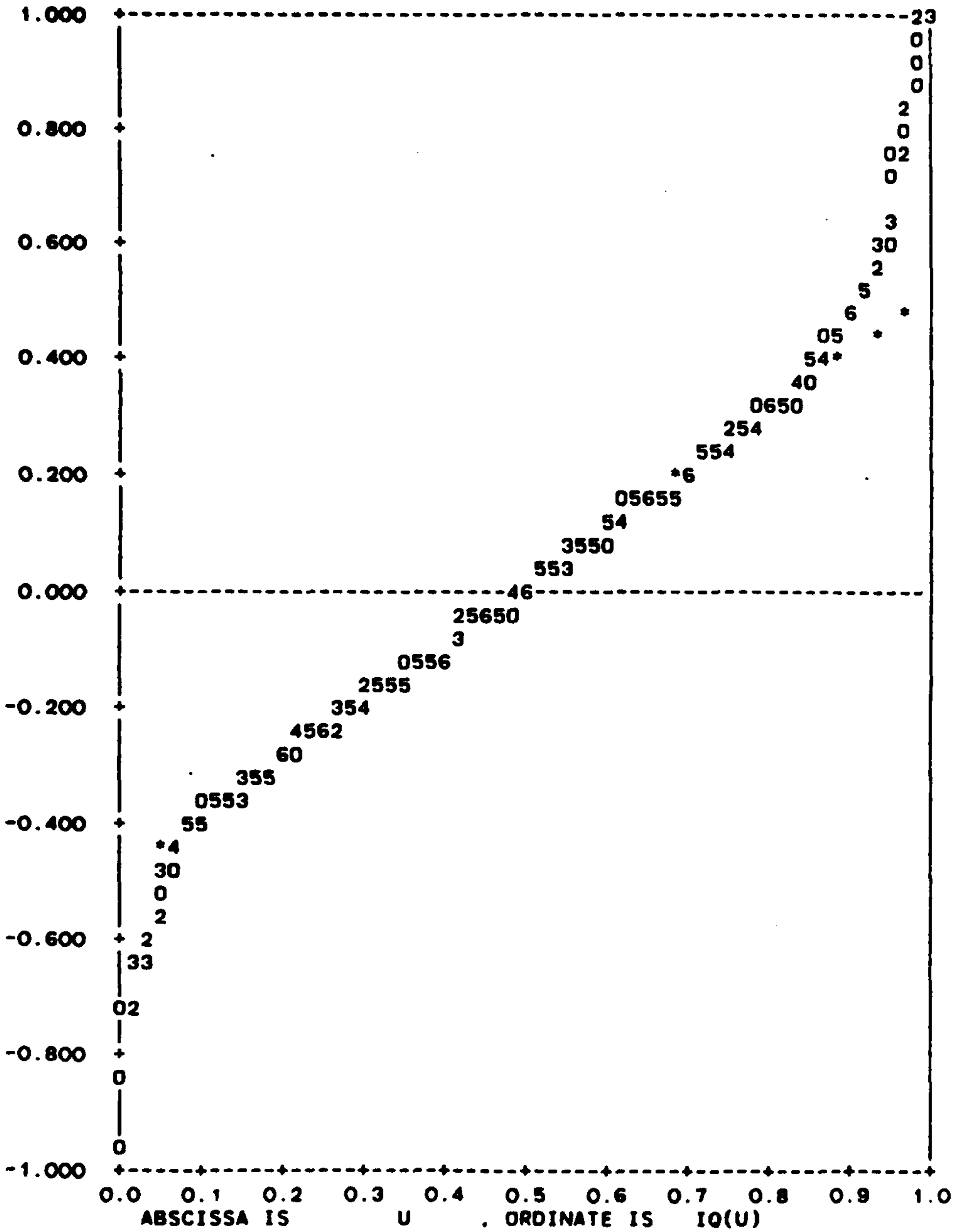
DESCRIPTIVE STATISTICS
****

SUMSTAT SUMSTAT	SAMPLE SIZE	LOWER QUARTILE	MEDIAN	UPPER QUARTILE	INT_QUARTL RANGE	TRIMEAN	GASTWIRTHS ESTIMATE
SUMSTAT	320	185.0	212.5	242.0	57.00	213.0	212.8

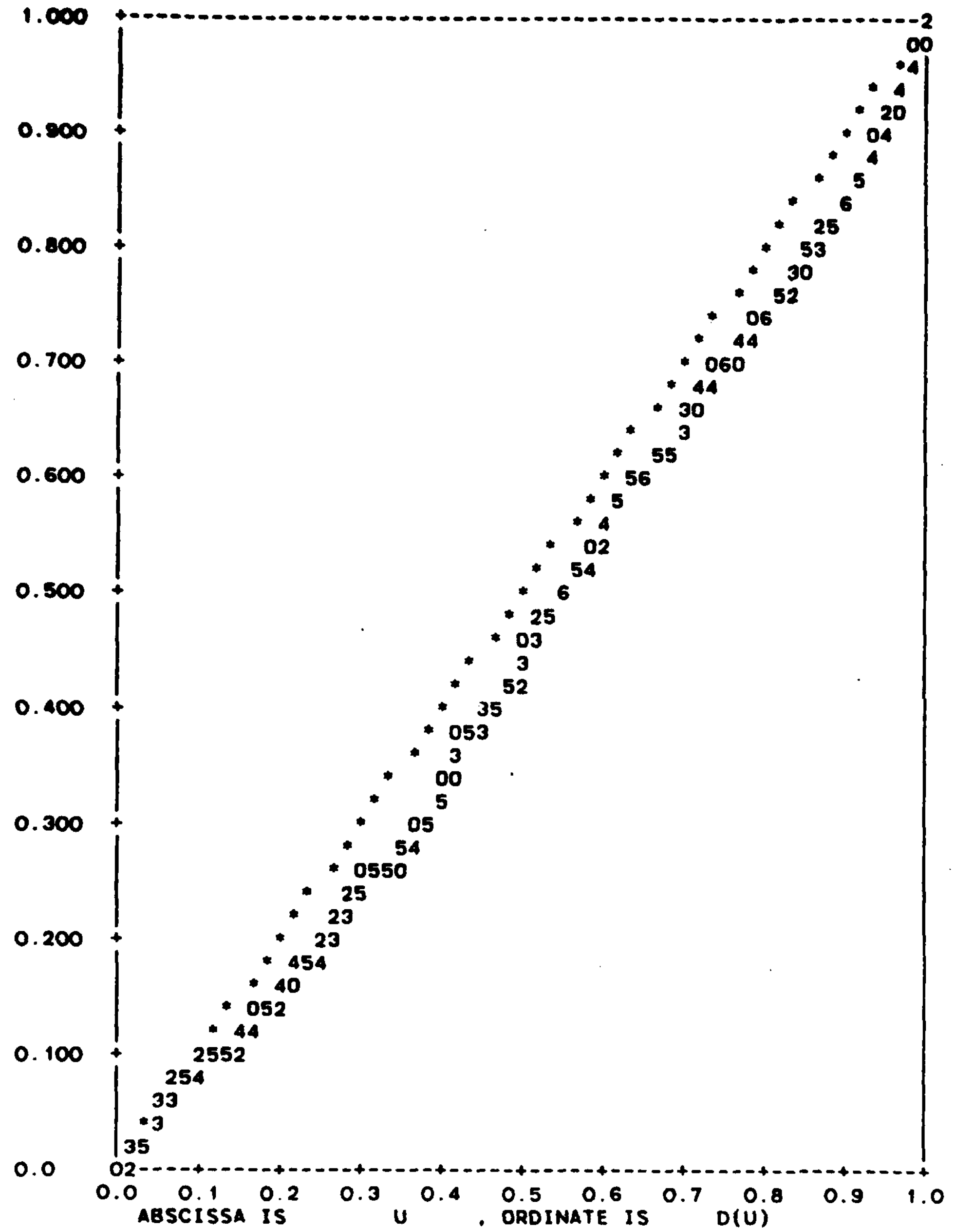
SUMSTAT	SUMSQ/N	MEAN	VARIANCE	STD DEV	MEAN IQ	STD DEV IQ	LOG STD IQ
SUMSTAT	.4858E+05	216.2	1850.	43.01	.3235E-01	.3773	-.9747

TRUNCATION POINT	WINSORIZED MEAN	TRIMMED MEAN
0.050	213.9	214.7
0.100	213.5	214.0
0.250	212.1	213.4

PLASMA CHOLESTEROL - DISEASE IN AT LEAST 1 OF 3 CORONARY ARTERIES
INFORMATIVE QUANTILE - ORIGINAL DATA - X



U	0.01000	0.05000	0.10000	0.25000	0.75000	0.90000	0.95000	0.99000
IQ(U)	-0.71491	-0.53684	-0.38947	-0.24123	0.25877	0.47807	0.65[illegible]63	1.15909


RAW DISTRIBUTION D(U) NORMAL CASE
D+ = 0.0360 AT U = 0.0516 . D- = -1.5590 AT U = 0.3859
1.000
0.900
0.800
0.700
0.600
0.500
0.400
0.300
0.200
0.100
0.0
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
ABSCISSA IS U , ORDINATE IS D(U)

FULLY NON-PARAMETRIC ANALYSIS
****

PLASMA TRIGLYCERIDES - DISEASE IN AT LEAST 1 OF 3 CORONARY ARTERIES
ORIGINAL DATA - Y

ORDER STATISTICS IN QUARTERS
****

SEQUENCE WITHIN QUARTILE	FIRST QUARTER	SECOND QUARTER	THIRD QUARTER	FOURTH QUARTER
1	36.0000	115.0000	150.0000	220.0000
2	38.0000	116.0000	151.0000	220.0000
3	50.0000	116.0000	151.0000	221.0000
4	54.0000	117.0000	151.0000	222.0000
5	56.0000	117.0000	152.0000	222.0000
6	59.0000	118.0000	152.0000	223.0000
7	61.0000	119.0000	152.0000	227.0000
8	68.0000	120.0000	152.0000	229.0000
9	72.0000	120.0000	153.0000	231.0000
10	73.0000	120.0000	153.0000	232.0000
11	75.0000	120.0000	153.0000	233.0000
12	76.0000	121.0000	153.0000	233.0000
13	77.0000	122.0000	154.0000	237.0000
14	78.0000	123.0000	154.0000	240.0000
15	80.0000	124.0000	154.0000	240.0000
16	80.0000	124.0000	154.0000	242.0000
17	80.0000	124.0000	155.0000	245.0000
18	80.0000	125.0000	156.0000	246.0000
19	82.0000	125.0000	156.0000	248.0000
20	82.0000	125.0000	156.0000	250.0000
21	84.0000	125.0000	156.0000	255.0000
22	84.0000	125.0000	158.0000	256.0000
23	84.0000	126.0000	158.0000	256.0000
24	84.0000	126.0000	158.0000	256.0000
25	84.0000	126.0000	160.0000	257.0000
26	85.0000	126.0000	161.0000	258.0000
27	87.0000	127.0000	161.0000	259.0000
28	87.0000	127.0000	161.0000	259.0000
29	88.0000	128.0000	162.0000	260.0000
30	88.0000	130.0000	162.0000	261.0000
31	89.0000	130.0000	163.0000	262.0000
32	89.0000	130.0000	164.0000	265.0000
33	90.0000	130.0000	164.0000	267.0000
34	90.0000	130.0000	165.0000	268.0000
35	90.0000	131.0000	166.0000	269.0000
36	91.0000	131.0000	166.0000	271.0000
37	91.0000	133.0000	168.0000	272.0000
38	91.0000	133.0000	168.0000	273.0000
39	91.0000	134.0000	169.0000	278.0000
40	91.0000	135.0000	170.0000	284.0000
41	92.0000	135.0000	170.0000	290.0000
42	92.0000	135.0000	170.0000	291.0000

43	92.0000	135.0000	170.0000	296.0000
44	93.0000	136.0000	171.0000	297.0000
45	95.0000	137.0000	172.0000	300.0000
46	96.0000	137.0000	172.0000	304.0000
47	96.0000	137.0000	173.0000	304.0000
48	97.0000	137.0000	174.0000	306.0000
49	98.0000	137.0000	176.0000	312.0000
50	99.0000	140.0000	177.0000	316.0000
51	100.0000	140.0000	179.0000	317.0000
52	100.0000	141.0000	179.0000	322.0000
53	101.0000	141.0000	180.0000	323.0000
54	101.0000	142.0000	181.0000	325.0000
55	101.0000	142.0000	182.0000	327.0000
56	101.0000	142.0000	183.0000	328.0000
57	101.0000	142.0000	184.0000	328.0000
58	101.0000	143.0000	188.0000	333.0000
59	102.0000	144.0000	189.0000	340.0000
60	102.0000	144.0000	192.0000	347.0000
61	103.0000	144.0000	195.0000	348.0000
62	103.0000	145.0000	196.0000	363.0000
63	104.0000	145.0000	196.0000	376.0000
64	105.0000	145.0000	198.0000	390.0000
65	106.0000	146.0000	199.0000	400.0000
66	107.0000	146.0000	199.0000	400.0000
67	108.0000	146.0000	199.0000	402.0000
68	108.0000	146.0000	200.0000	408.0000
69	108.0000	146.0000	201.0000	418.0000
70	109.0000	148.0000	201.0000	424.0000
71	110.0000	148.0000	202.0000	426.0000
72	110.0000	148.0000	202.0000	432.0000
73	111.0000	148.0000	207.0000	441.0000
74	112.0000	148.0000	207.0000	446.0000
75	112.0000	149.0000	208.0000	454.0000
76	112.0000	149.0000	209.0000	489.0000
77	112.0000	149.0000	210.0000	492.0000
78	115.0000	149.0000	217.0000	567.0000
79	115.0000	149.0000	217.0000	583.0000
80	115.0000	150.0000	218.0000	930.0000
SUM	7260.0000	10725.0000	13965.0000	25442.0000
SUM OF SQUARES	682086.000	1446605.00	2469517.00	9003904.00

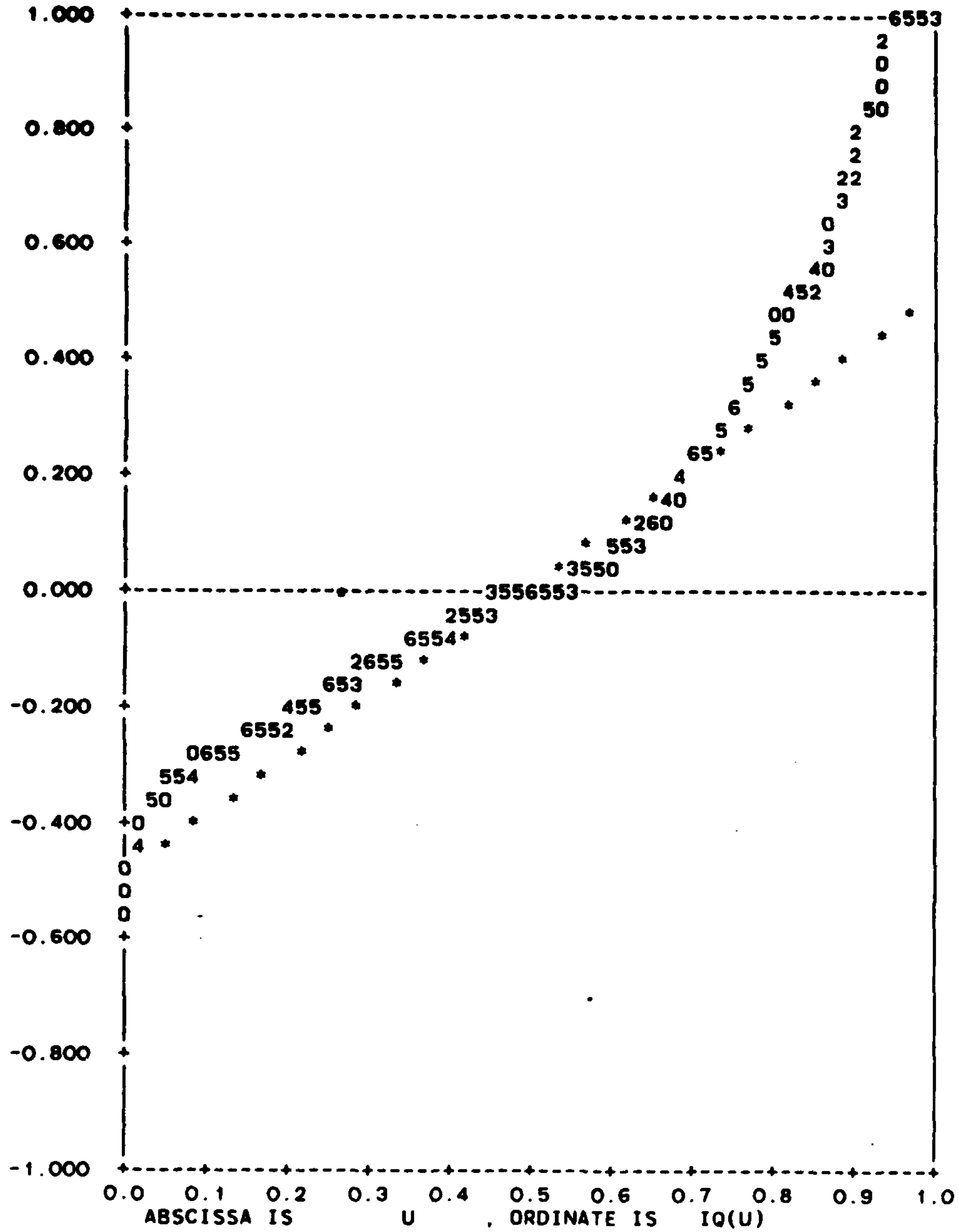
DESCRIPTIVE STATISTICS
****

SUMSTAT SUMSTAT	SAMPLE SIZE	LOWER QUARTILE	MEDIAN	UPPER QUARTILE	INT_QUARTL RANGE	TRIMEAN	GASTWIRTHS ESTIMATE
SUMSTAT	320	115.0	150.0	219.5	104.5	158.6	152.4

SUMSTAT	SUMSQ/N	MEAN	VARIANCE	STD DEV	MEAN IQ	STD DEV IQ	LOG STD IQ
SUMSTAT	.4251E+05	179.3	.1037E+05	101.8	.1404	.4873	-.7189

TRUNCATION POINT	WINSORIZED MEAN	TRIMMED MEAN
0.050	174.1	168.9
0.100	169.3	163.7
0.250	159.4	154.3

PLASMA TRIGLYCERIDES - DISEASE IN AT LEAST 1 OF 3 CORONARY ARTERIES
INFORMATIVE QUANTILE - ORIGINAL DATA - Y

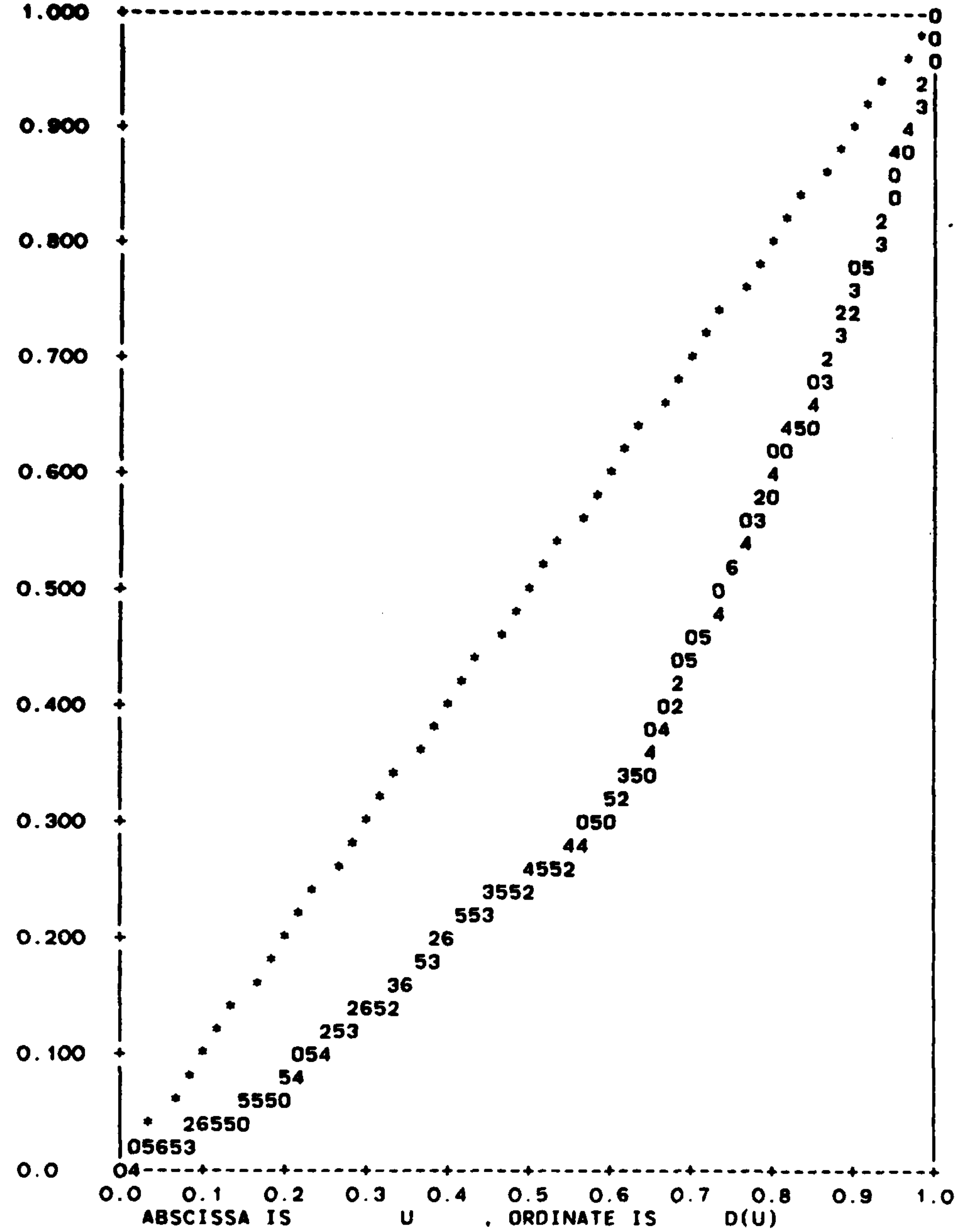


U	0.01000	0.05000	0.10000	0.25000	0.75000	0.90000	0.95000	0.99000
IQ(U)	-0.47445	-0.33493	-0.29139	-0.16746	0.33254	0.77225	1.19378	1.91978

This page is blank.

RAW DISTRIBUTION D(U) NORMAL CASE

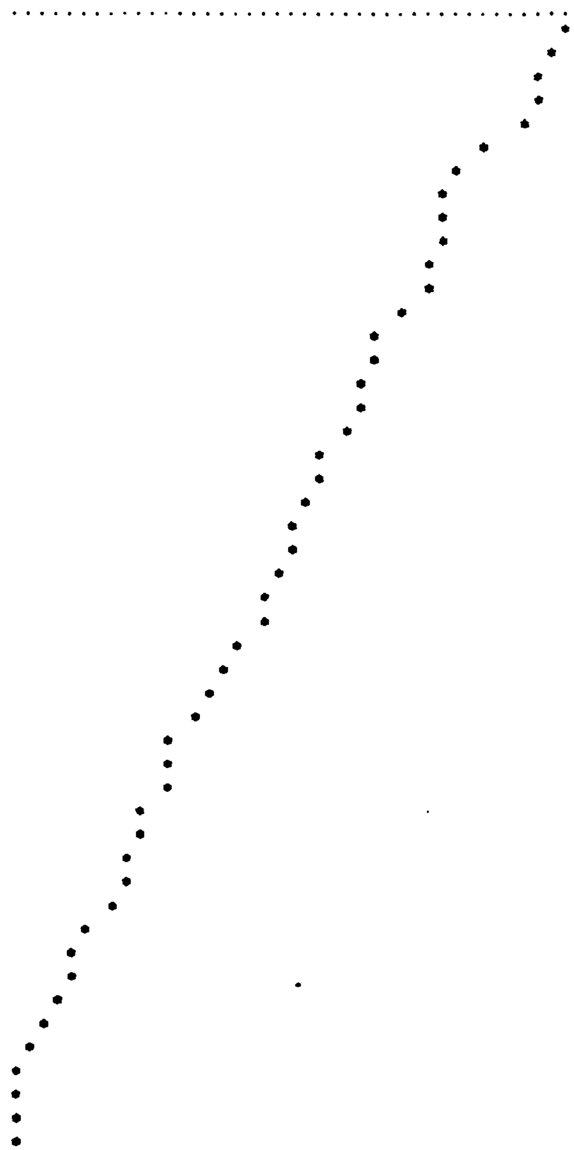
D+ = 0.0282 AT U = 0.9984 . D- = -5.2744 AT U = 0.6422



THE FOLLOWING POINTS WERE DELETED FROM THE DATA SET:
320

1 POINTS WERE DELETED LEAVING 319 POINTS IN THE DATA SET.

I	RVAR
1	0.9905
2	0.9799
3	0.9646
4	0.9562
5	0.9457
6	0.9033
7	0.8624
8	0.8471
9	0.8434
10	0.8424
11	0.8382
12	0.8316
13	0.8071
14	0.7790
15	0.7660
16	0.7636
17	0.7621
18	0.7385
19	0.7115
20	0.7092
21	0.7002
22	0.6845
23	0.6702
24	0.6571
25	0.6489
26	0.6402
27	0.6133
28	0.6061
29	0.5895
30	0.5632
31	0.5417
32	0.5353
33	0.5292
34	0.5037
35	0.5025
36	0.4944
37	0.4907
38	0.4792
39	0.4503
40	0.4308
41	0.4229
42	0.4191
43	0.3909
44	0.3737
45	0.3699
46	0.3679
47	0.3655
48	0.3649



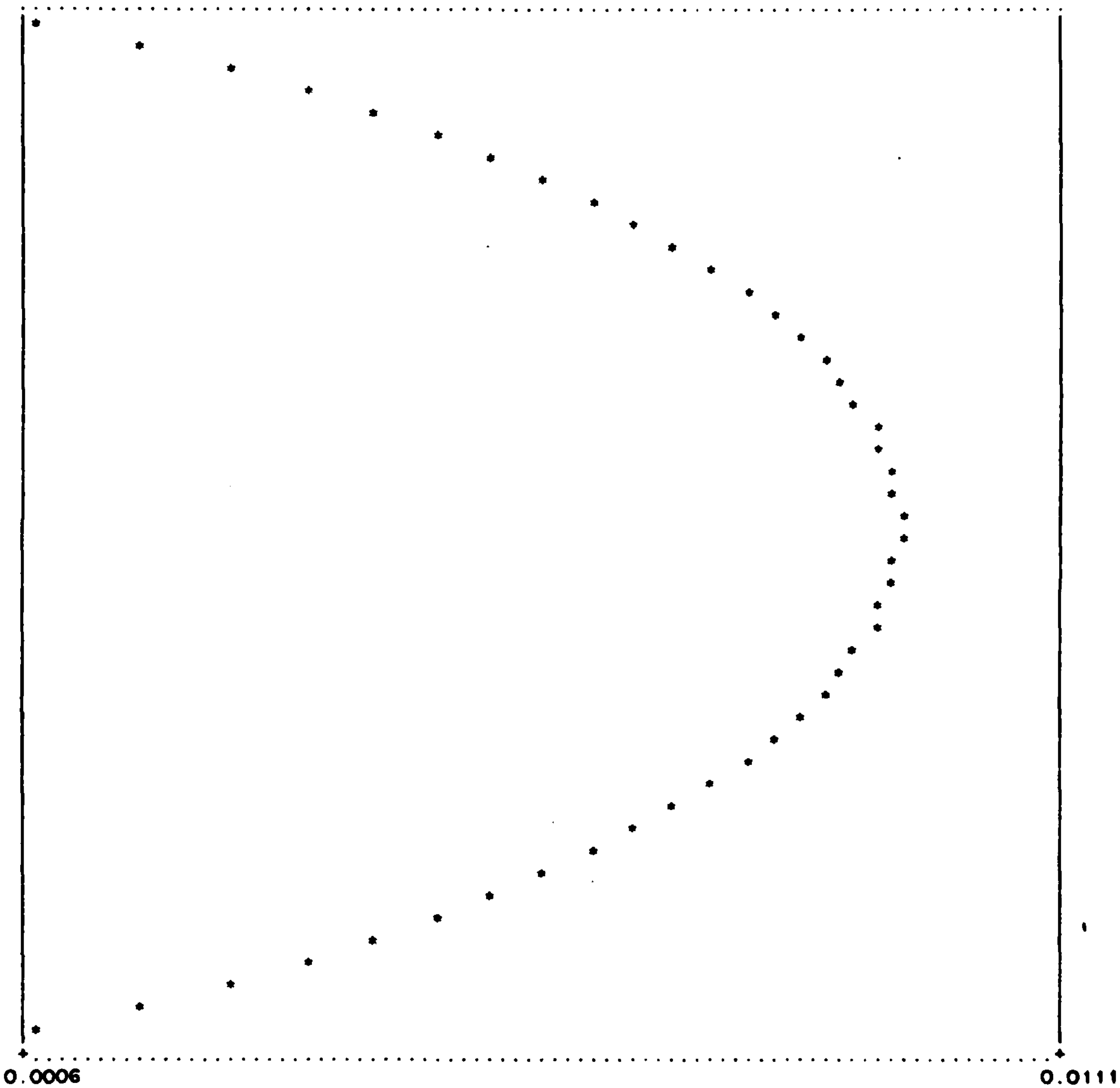
UNIVARIATE DENSITY ESTIMATION RESULTS FOR VARIABLE X

I	RVAR
1	0.9972
2	0.9876

SIGO = 42.6821

UNIVARIATE DENSITY-QUANTILE FOR RANDOM VARIABLE X

U	FQ
0.0099	0.0006
0.0317	0.0017
0.0535	0.0025
0.0752	0.0033
0.0970	0.0040
0.1188	0.0047
0.1406	0.0052
0.1624	0.0058
0.1842	0.0062
0.2059	0.0067
0.2277	0.0071
0.2495	0.0074
0.2713	0.0078
0.2931	0.0081
0.3149	0.0083
0.3366	0.0086
0.3584	0.0088
0.3802	0.0089
0.4020	0.0091
0.4238	0.0092
0.4455	0.0093
0.4673	0.0093
0.4891	0.0093
0.5109	0.0093
0.5327	0.0093
0.5545	0.0093
0.5762	0.0092
0.5980	0.0091
0.6198	0.0089
0.6416	0.0088
0.6634	0.0086
0.6851	0.0083
0.7069	0.0081
0.7287	0.0078
0.7505	0.0074
0.7723	0.0071
0.7941	0.0067
0.8158	0.0062
0.8376	0.0058
0.8594	0.0052
0.8812	0.0047
0.9030	0.0040
0.9248	0.0033
0.9465	0.0025
0.9683	0.0017
0.9901	0.0006



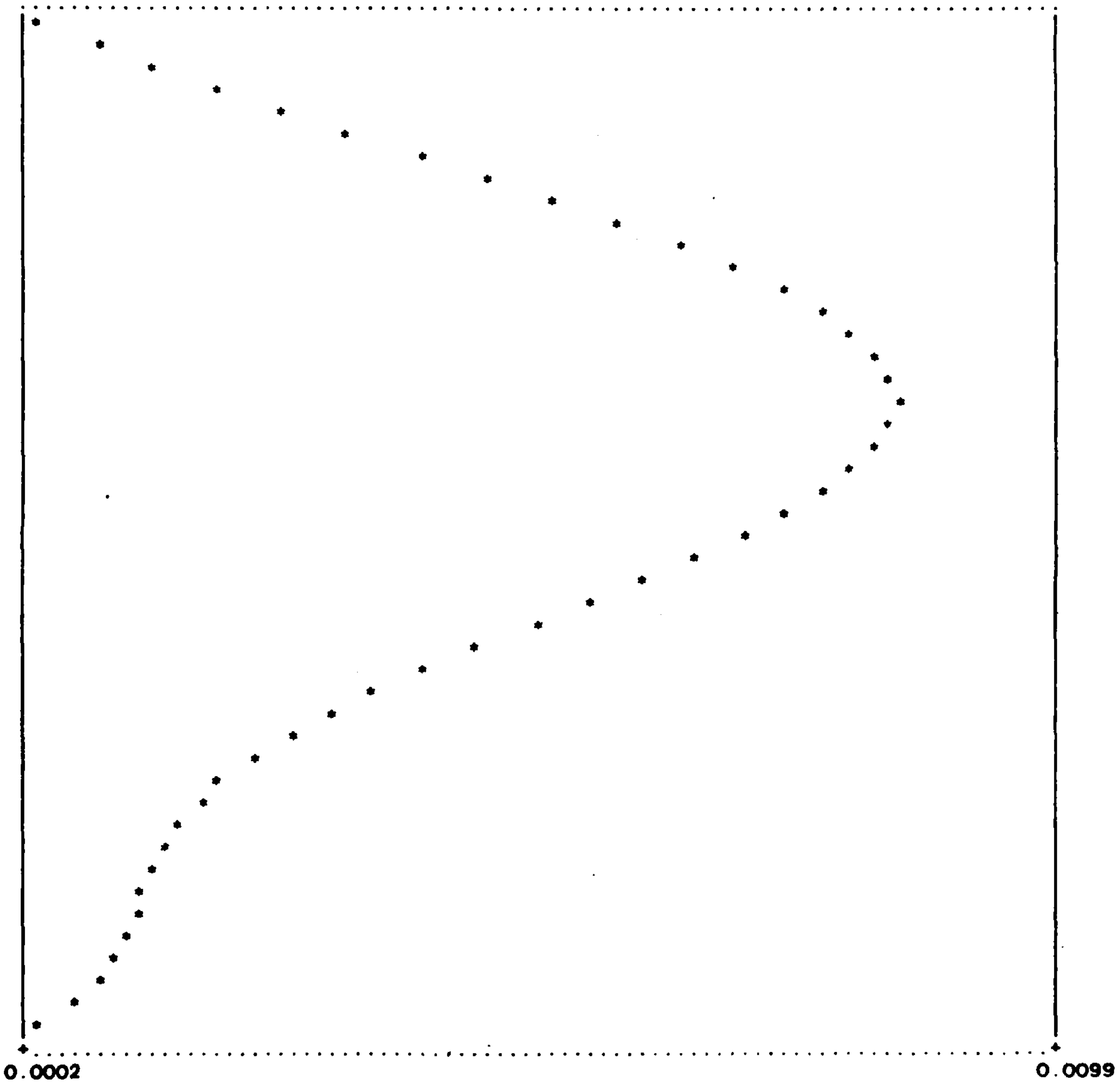
```
UNIVARIATE DENSITY ESTIMATION RESULTS FOR VARIABLE   Y


    I      RVAR
---------------  ..........................................
    1    0.8826                                           *
    2    0.8789  *

SIGO =    91.7530
```

UNIVARIATE DENSITY-QUANTILE FOR RANDOM VARIABLE Y

U	FQ
0.0099	0.0003
0.0317	0.0008
0.0535	0.0014
0.0752	0.0019
0.0970	0.0026
0.1188	0.0032
0.1406	0.0038
0.1624	0.0045
0.1842	0.0051
0.2059	0.0057
0.2277	0.0062
0.2495	0.0068
0.2713	0.0072
0.2931	0.0076
0.3149	0.0079
0.3366	0.0081
0.3584	0.0083
0.3802	0.0083
0.4020	0.0083
0.4238	0.0081
0.4455	0.0079
0.4673	0.0076
0.4891	0.0073
0.5109	0.0069
0.5327	0.0064
0.5545	0.0059
0.5762	0.0054
0.5980	0.0049
0.6198	0.0044
0.6416	0.0039
0.6634	0.0034
0.6851	0.0030
0.7069	0.0026
0.7287	0.0023
0.7505	0.0020
0.7723	0.0018
0.7941	0.0016
0.8158	0.0014
0.8376	0.0013
0.8594	0.0012
0.8812	0.0012
0.9030	0.0011
0.9248	0.0010
0.9465	0.0008
0.9683	0.0006
0.9901	0.0002



UNIVARIATE BEST ORDERS: NVX = 0, NVY = 1

RESULTS FOR ORDER 8 MODEL:

U1	U2	DOHT	DHAT
0.18749994	0.18749994	0.00004772	1.46289539
0.18749994	0.38749993	0.00005591	1.06929111
0.18749994	0.58749992	0.00002575	0.79277152
0.18749994	0.78749990	0.00000923	0.90147328
0.18749994	0.98749989	0.00000250	1.31640625
0.38749993	0.18749994	0.00006505	1.40138245
0.38749993	0.38749993	0.00009387	1.26168442
0.38749993	0.58749992	0.00004522	0.97862923
0.38749993	0.78749990	0.00001353	0.92904085
0.38749993	0.98749989	0.00000313	1.15987206
0.58749992	0.18749994	0.00004023	0.85252440
0.58749992	0.38749993	0.00008944	1.18259430
0.58749992	0.58749992	0.00006110	1.30061340
0.58749992	0.78749990	0.00001472	0.99437904
0.58749992	0.98749989	0.00000210	0.76591164
0.78749990	0.18749994	0.00002302	0.65457690
0.78749990	0.38749993	0.00005429	0.96295321
0.78749990	0.58749992	0.00004399	1.25609970
0.78749990	0.78749990	0.00001111	1.00626469
0.78749990	0.98749989	0.00000138	0.67261958
0.98749989	0.18749994	0.00000358	0.91388923
0.98749989	0.38749993	0.00000569	0.90484190
0.98749989	0.58749992	0.00000361	0.92501044
0.98749989	0.78749990	0.00000117	0.94707549
0.98749989	0.98749989	0.00000021	0.94002223

INTEGRATING FACTOR FOR ORDER 8 IS 0.9728

MAXIMUM VALUE FOR DEPENDENCE DENSITY QUANTILE:

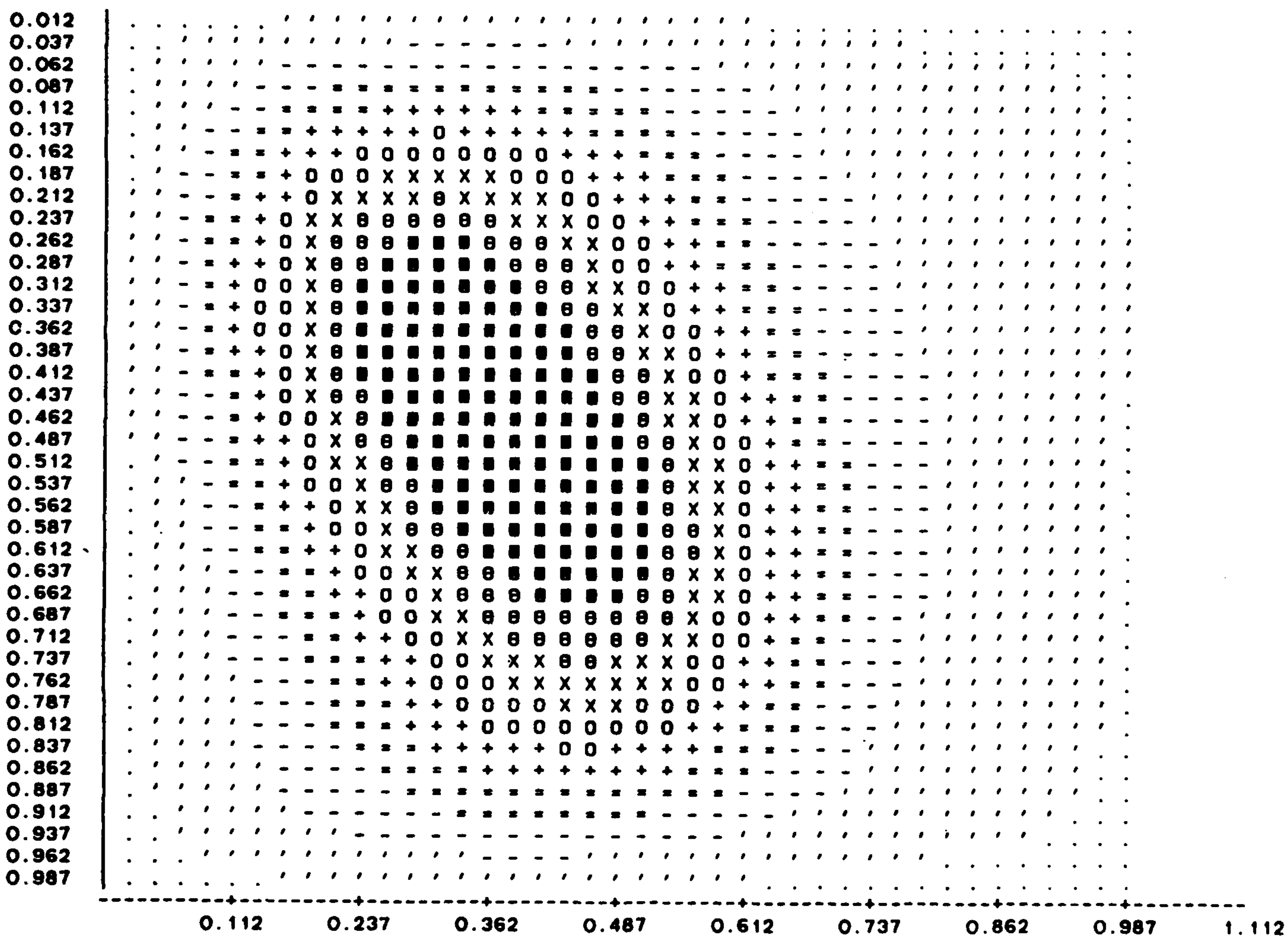
D(0.43750,0.36250) = 0.0001011

U1*N = 139.56 U2*N = 115.64

COEFFICIENTS FOR BIVARIATE DEPENDENCE DENSITY

NU1	NU2	REAL(COF)	IMAG(COF)
0	-1	-0.0290	0.0168
-1	0	-0.0341	0.0560
-1	-1	-0.0360	0.0561
0	1	-0.0290	-0.0168
1	0	-0.0341	-0.0560
-1	1	0.0820	0.0736
1	-1	0.0820	-0.0736
1	1	-0.0360	-0.0561

PLASMA CHOLESTEROL - DISEASE IN AT LEAST 1 OF 3 CORONARY ARTERIES
PLASMA TRIGLYCERIDES - DISEASE IN AT LEAST 1 OF 3 CORONARY ARTERIES
CONTOUR PLOT FOR BIVARIATE DENSITY QUANTILE - ORDER = 8



ORDINATE IS U1, ABSCISSA IS U2
U1 CORRESPONDS TO X (FIRST VARIABLE), U2 TO Y

RESULTS FOR ORDER 24 MODEL:

U1	U2	DOHT	DHAT
0.18749994	0.18749994	0.00006591	2.02025032
0.18749994	0.38749993	0.00005909	1.13007641
0.18749994	0.58749992	0.00002053	0.63217348
0.18749994	0.78749990	0.00000841	0.82144260
0.18749994	0.98749989	0.00000127	0.67156988
0.38749993	0.18749994	0.00005310	1.14403343
0.38749993	0.38749993	0.00008828	1.18650246
0.38749993	0.58749992	0.00004822	1.04348946
0.38749993	0.78749990	0.00002334	1.60251522
0.38749993	0.98749989	0.00000298	1.10544777
0.58749992	0.18749994	0.00004044	0.85700732
0.58749992	0.38749993	0.00012231	1.61717510
0.58749992	0.58749992	0.00005357	1.14039707
0.58749992	0.78749990	0.00001090	0.73636955
0.58749992	0.98749989	0.00000271	0.98877615
0.78749990	0.18749994	0.00002065	0.58699453
0.78749990	0.38749993	0.00006526	1.15753841
0.78749990	0.58749992	0.00004041	1.15400219
0.78749990	0.78749990	0.00001366	1.23773098
0.78749990	0.98749989	0.00000131	0.64088982
0.98749989	0.18749994	0.00000330	0.84250641
0.98749989	0.38749993	0.00000408	0.64825362
0.98749989	0.58749992	0.00000514	1.31678867
0.98749989	0.78749990	0.00000102	0.82501429
0.98749989	0.98749989	0.00000027	1.17862511

==

INTEGRATING FACTOR FOR ORDER 24 IS 1.0001

MAXIMUM VALUE FOR DEPENDENCE DENSITY QUANTILE:

D(0.61250,0.41250) = 0.0001278

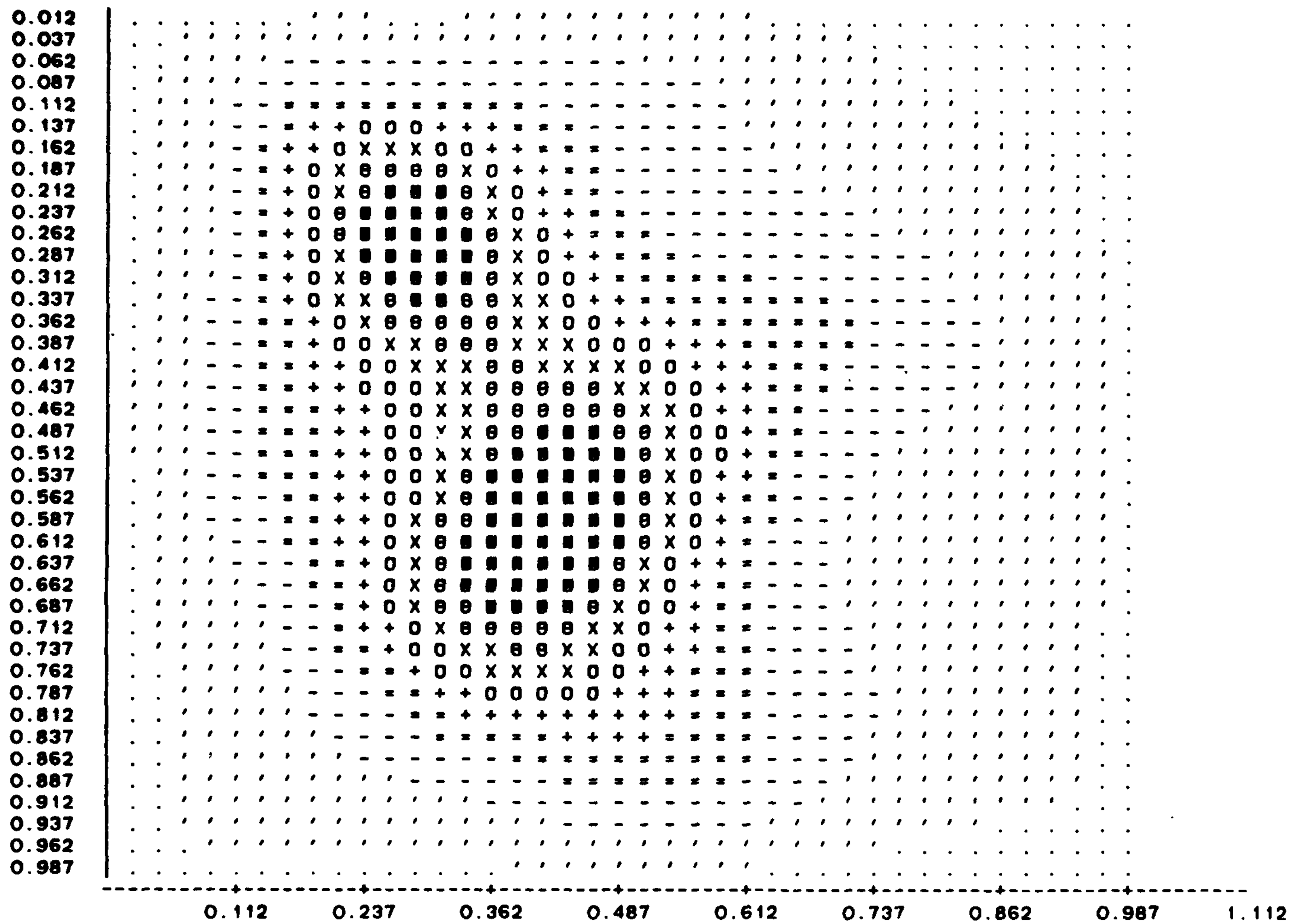
U1*N = 195.39 U2*N = 131.59

COEFFICIENTS FOR BIVARIATE DEPENDENCE DENSITY

NU1	NU2	REAL(COF)	IMAG(COF)
0	-1	-0.0433	0.0113
-1	0	-0.0550	0.0321
-1	-1	-0.0372	0.0418
0	1	-0.0433	-0.0113
1	0	-0.0550	-0.0321
-1	1	0.0737	0.0619
1	-1	0.0737	-0.0619
1	1	-0.0372	-0.0418
0	-2	-0.0169	-0.0112
-2	0	0.0175	-0.0269
-1	-2	-0.0197	-0.0072
-2	-1	-0.0509	-0.0294
1	-2	0.0178	0.0904

-2	1	0.0935	0.0131
-2	-2	0.0525	0.0351
0	2	-0.0169	0.0112
2	0	0.0175	0.0269
-1	2	0.0178	-0.0904
2	-1	0.0935	-0.0131
1	2	-0.0197	0.0072
2	1	-0.0509	0.0294
-2	2	0.0646	-0.0214
2	-2	0.0646	0.0214
2	2	0.0525	-0.0351

PLASMA CHOLESTEROL - DISEASE IN AT LEAST 1 OF 3 CORONARY ARTERIES
PLASMA TRIGLYCERIDES - DISEASE IN AT LEAST 1 OF 3 CORONARY ARTERIES
CONTOUR PLOT FOR BIVARIATE DENSITY QUANTILE - ORDER = 24



ORDINATE IS U1, ABSCISSA IS U2
U1 CORRESPONDS TO X (FIRST VARIABLE), U2 TO Y

RESULTS FOR ORDER 48 MODEL:

U1	U2	DQHT	DHAT
0.18749994	0.18749994	0.00006638	2.03462219
0.18749994	0.38749993	0.00005525	1.05659676
0.18749994	0.58749992	0.00003006	0.92565721
0.18749994	0.78749990	0.00000834	0.81518406
0.18749994	0.98749989	0.00000080	0.42414057
0.38749993	0.18749994	0.00010054	2.16585255
0.38749993	0.38749993	0.00006806	0.91477740
0.38749993	0.58749992	0.00005214	1.12818909
0.38749993	0.78749990	0.00001949	1.33813477
0.38749993	0.98749989	0.00000247	0.91500378
0.58749992	0.18749994	0.00002933	0.62150997
0.58749992	0.38749993	0.00010528	1.39199924
0.58749992	0.58749992	0.00005599	1.19188213
0.58749992	0.78749990	0.00001759	1.18802643
0.58749992	0.98749989	0.00000212	0.77384871
0.78749990	0.18749994	0.00003154	0.89657325
0.78749990	0.38749993	0.00006950	1.23278046
0.78749990	0.58749992	0.00004922	1.40551186
0.78749990	0.78749990	0.00001147	1.03926563
0.78749990	0.98749989	0.00000134	0.65280330
0.98749989	0.18749994	0.00000217	0.55329788
0.98749989	0.38749993	0.00000563	0.89526796
0.98749989	0.58749992	0.00000369	0.94539171
0.98749989	0.78749990	0.00000111	0.90271294
0.98749989	0.98749989	0.00000029	1.25734138

==

INTEGRATING FACTOR FOR ORDER 48 IS · 1.0443

MAXIMUM VALUE FOR DEPENDENCE DENSITY QUANTILE:

D(0.43750,0.48750) = 0.0001421

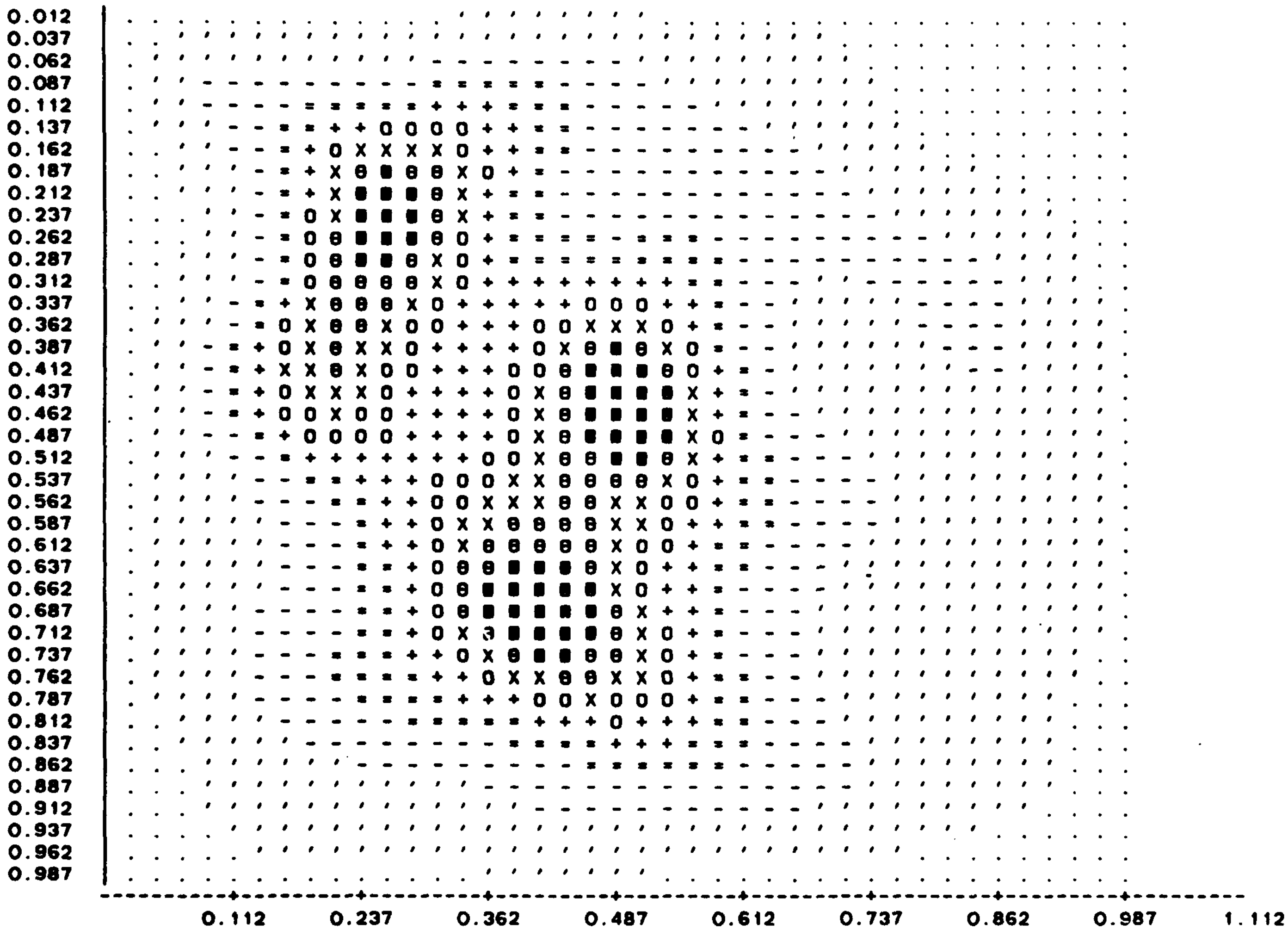
U1*N = 139.56 U2*N = 155.51

COEFFICIENTS FOR BIVARIATE DEPENDENCE DENSITY

NU1	NU2	REAL(COF)	IMAG(COF)
0	-1	-0.0785	0.0079
-1	0	-0.0592	0.0134
-1	-1	-0.0556	0.0353
0	1	-0.0785	-0.0079
1	0	-0.0592	-0.0134
-1	1	0.0687	0.0356
1	-1	0.0687	-0.0356
1	1	-0.0556	-0.0353
0	-2	-0.0196	-0.0085
-2	0	0.0075	-0.0215
-1	-2	-0.0236	-0.0120
-2	-1	-0.0386	-0.0357
1	-2	0.0194	0.1037

-2	1	0.0895	0.0027
-2	-2	0.0487	0.0302
0	2	-0.0196	0.0085
2	0	0.0075	0.0215
-1	2	0.0194	-0.1037
2	-1	0.0895	-0.0027
1	2	-0.0236	0.0120
2	1	-0.0386	0.0357
-2	2	0.0708	-0.0040
2	-2	0.0708	0.0040
2	2	0.0487	-0.0302
0	-3	-0.0462	-0.0077
-3	0	-0.0064	0.0327
-1	-3	0.0733	0.0192
-3	-1	-0.0524	0.0136
1	-3	0.0271	0.0480
-3	1	0.0891	0.0206
-2	-3	-0.0161	0.0730
-3	-2	-0.0033	0.0341
2	-3	0.0262	-0.0330
-3	2	0.0512	0.0774
-3	-3	-0.0111	0.0080
0	3	-0.0462	0.0077
3	0	-0.0064	-0.0327
-1	3	0.0271	-0.0480
3	-1	0.0891	-0.0206
1	3	0.0733	-0.0192
3	1	-0.0524	-0.0136
-2	3	0.0262	0.0330
3	-2	0.0512	-0.0774
2	3	-0.0161	-0.0730
3	2	-0.0033	-0.0341
-3	3	0.0259	-0.0136
3	-3	0.0259	0.0136
3	3	-0.0111	-0.0080

PLASMA CHOLESTEROL - DISEASE IN AT LEAST 1 OF 3 CORONARY ARTERIES
PLASMA TRIGLYCERIDES - DISEASE IN AT LEAST 1 OF 3 CORONARY ARTERIES
CONTOUR PLOT FOR BIVARIATE DENSITY QUANTILE - ORDER = 48



ORDINATE IS U1, ABSCISSA IS U2
U1 CORRESPONDS TO X (FIRST VARIABLE), U2 TO Y

BEST MODEL BY AIC IS ORDER 8 MODEL.

TIES IN X = 388, TIES IN Y = 234

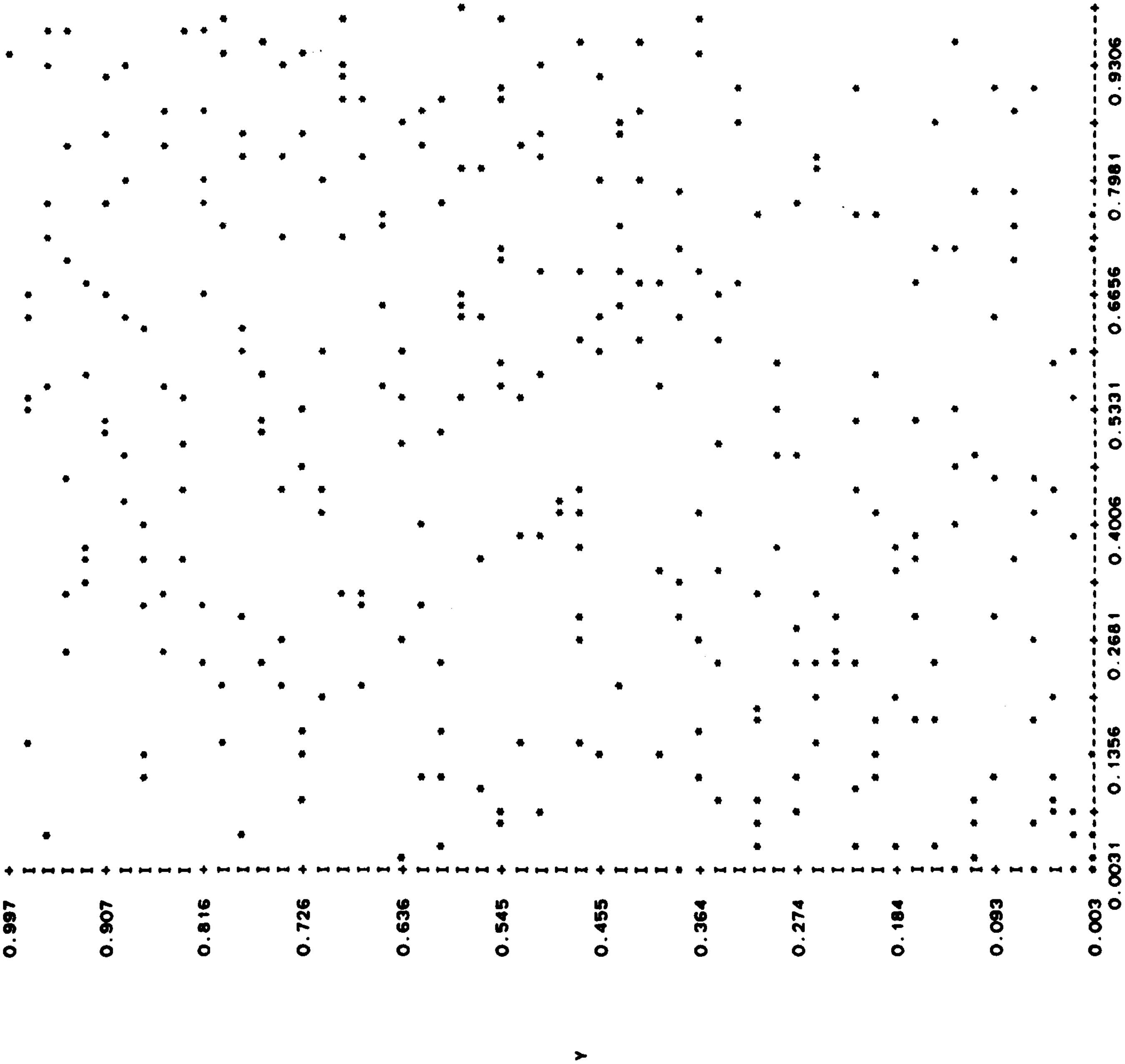
PLASMA CHOLESTEROL - DISEASE IN AT LEAST 1 OF 3 CORONARY ARTERIES
PLASMA TRIGLYCERIDES - DISEASE IN AT LEAST 1 OF 3 CORONARY ARTERIES

SAMPLE SIZE = 319

CORRELATION COEFICIENT	VALUE	INFORMATION (NORMAL CASE)
PEARSON	0.233332	0.027991
SPEARMAN	0.273826	0.038970
KENDALL A	0.186018	0.017608
KENDALL B	0.187173	0.017831
SOMER'S D	0.187459	0.017887

MODEL	INFORMATION	AIC
D-TILDA	0.03909	0.0
D-HAT 8	0.07118	-0.08224
D-HAT 24	0.09854	-0.20992
D-HAT 48	0.13924	-0.40108

SCATTER PLOT OF X VS. Y - RANK TRANSFORMATION



APPENDIX A - JCL for Executing BISAM

```
// Job Card
//* JES3 Control Cards
//PROCLIB DD DSN=USR.R579.TW.PROCLIB,DISP=SHR
// EXEC BISAM
//SYSIN DD *
      Parameter Input Card(s)
      Data if NTAPE=5
```

If NTAPE≠5, the value of NTAPE is coded as a number nn between 8 and 99 except for 11. (The value 11 is currently used for a scratch file.) A DD card is then inserted defining the input data set. For example,

```
//FT13FØØ1 DD DSN=USR.R579.TW.DATA,DISP-SHR
```

described the dish file containing the input data with NTAPE=13. If IOUTD-1, three such cards will be required as described in section 6.

APPENDIX B - OBTAINING THREE DIMENSIONAL PLOTS

```
 1.   //  JOB CARD
 2.   //* JES3 CONTROL CARDS
 3.   //PROCLIB DD DSN=USR.R579.TW.PROCLIB,DISP=SHR
 4.   // EXEC BISAM
 5.   //SYSIN DD *
 6.
 7.
 8.
 9.
10.   *** OPTION CARD FOLLOWED BY DATA IF NTAPE=5 ***
11.   //FT01F001 DD DSN=WYL.XX.YYY.STOR1,DISP=OLD
12.   //FT02F001 DD DSN=WYL.XX.YYY.STOR2,DISP=OLD
13.   //FT03F001 DD DSN=WYL.XX.YYY.STOR3,DISP=OLD
14.   //FT13F001 DD DSN=WYL.XX.YYY.DAT1,DISP=OLD
15.   /*
16.   //STEP2 EXEC SAS,COND=(0,LT)
17.   //FILE1 DD DSN=WYL.XX.YYY.STOR1,DISP=OLD
18.   //FILE2 DD DSN=WYL.XX.YYY.STOR2,DISP=OLD
19.   //FILE3 DD DSN=WYL.XX.YYY.STOR3,DISP=OLD
20.   //SYSIN DD *
21.   DATA ONE; INFILE FILE1; INPUT U1 U2 DQHT DHAT;
22.   TITLE .F=COMPLEX .H=1 FIRST ORDER BIVARIATE DENSITY;
23.   PROC G3D DATA=ONE GOUT=A; PLOT U1*U2=DQHT;.
24.   PROC G3D DATA=ONE GOUT=B; PLOT U1*U2=DHAT;
25.   DATA TWO; INFILE FILE2; INPUT U1 U2 DQHT DHAT;
26.   TITLE .F=COMPLEX .H=1 SECOND ORDER BIVARIATE DENSITY;
27.   PROC G3D DATA=TWO GOUT=C; PLOT U1*U2=DQHT;
28.   PROC G3D DATA=TWO GOUT=D; PLOT U1*U2=DHAT;
29.   DATA THREE; INFILE FILE3; INPUT U1 U2 DQHT DHAT;
30.   TITLE .F=COMPLEX .H=1 THIRD ORDER BIVARIATE DENSITY;
31.   PROC G3D DATA=THREE GOUT=E; PLOT U1*U2=DQHT;
32.   PROC G3D DATA=THREE GOUT=F; PLOT U1*U2=DHAT;
33.   DATA COMBINE; SET  A B C D E F;
34.   PROC GREPLAY DATA=COMBINE;
35.   /*
```

NOTES: 1. USE CURRENT SAS/GRAPH JCL IN LINE 16.

3. XX.YYY IS USER'S ACCOUNT.

5. THE SAS PROCEDURE GCONTOUR MAY BE SUBSTITUTED FOR G3D TO PRODUCE CONTOUR PLOTS INSTEAD OF 3D PLOTS.
6. AS INDICATED, PLOTS OF DHAT MAY NOT BE INFORMATIVE, SO DHAT PLOTS INDICATED ABOVE ARE USUALLY OMITTED.
7. IN LINES 11-13, THE WYLBUR FILES ARE DUMMY FILES SAVED IN CARD IMAGE THAT WILL BE WRITTEN TO. IN LINE 14, NTAPE=13 HAS BEEN SPECIFIED AND DAT1 CONTAINS THE BIVARIATE DATA SET TO BE ANALYZED.

APPENDIX C - PROGRAM LISTING

MEMBER	NUMBER OF RECORDS
BISAM	2706

```
C     PROGRAM BISAM
C
C******************************************************
C
C   DRIVER PROGRAM FOR BIVARIATE DATA ANALYSIS
C
C   INPUT: NTAPE - TAPE WHERE DATA SET RESIDES
C          (X,Y) - BIVARIATE DATA (INDIVIDUALLY, X FIRST)
C          MORD - MAXIMUM AUTOREGRESSIVE ORDER TO BE USED FOR
C                 UNIVARIATE AR DENSITY ESTIMATION (<=6)
C          IDOX,IDOY - NULL DISTRIBUTIONS FOR AUTOREGRESSIVE SMOOTHING
C          IPLT1 - 0--> NO SCATTER PLOTS
C                  1--> SCATTER PLOT OF DATA
C                  2--> SCATTER PLOT OF RANK TRANSFORMED DATA
C                  3--> BOTH SCATTER PLOTS
C          IPLT2 - 0--> NO AUTOREGRESSIVE DENSITY PLOTS
C                  1--> BEST ORDER AR DENSITY PLOTS
C          IDST - 0--> NO UNIVARIATE DESCRIPTIVE STATISTICS
C                 1--> UNIVARIATE DESCRIPTIVE STATISTICS FOR X AND Y
C          KDEL - MAXIMUM NUMBER OF EXTREME POINTS TO EXCLUDE FROM
C                 BIVARIATE ANALYSIS
C          IOUTD - 1 IF THE 3 MODELS ESTIMATED ARE TO HAVE VALUES
C                  WRITTEN TO TAPES 1,2, AND 3; 0 O.W
C                  FT01001, ETC. DD JCL CARDS MUST BE INCLUDED IF
C                  IOUTD=1.
C          IREG - 1 IF QUANTILE REGRESSION PERFORMED, 0 O.W.
C          IUNIV - 1 IF UNIVARIATE DATA SET TO BE READ IN AND A
C                  BIVARIATE DATA SET CREATED AS ORIGINAL DATA
C                  AND LAGGED DATA, 0 O.W.
C          KLAG1,KLAG2 - RANGE OF LAGS INPUT IF IUNIV=1
C
C   SUBPROGRAMS CALLED: AREST,AUTOEN,AUTORG,CLPLT1,CMPINF,CPLOT,
C                       CPTENT,CQOREG,CSWEEP,DATAIN,DESTAT,FCODEA,
C                       FORIER,FOPHC,FTERP,ICODEA,KENDAL,KSD,MAX,
C                       MONRIS,MIN,MINMAX,ORD2,PARZ,PEARSN,PLOTXY,
C                       PPLOT,QFIND,QPLOT,QRES,QTOFQ,QUENT,QUICK,
C                       RANK,SPRMN,TRIM,WSPACE
C
C   NOTE: SCRATCH TAPE NUMBER IS SET AT 11 IN FCODEA. A DD CARD
C         FOR FT11F001 MUST BE INCLUDED IN THE JCL SET UP.
C
C   JULY 1983
C   PROGRAMMER: TERRY J. WOODFIELD
C
C******************************************************
C
      COMMON /DATAR/ X(500),Y(500),RANKX(500),RANKY(500)
      COMMON /PARM/ BETAP,BETAW
      DIMENSION L(8),LABY(20),LABX(20),T(500,2),MD(4),AIC(4)
      DIMENSION CHAR(5),XNAME(20),YNAME(20),CAPT(20),CRNK(6)
      DIMENSION W(1000),LABO(2),LABDX(20),LABDY(20)
      EQUIVALENCE (T(1,1),X(1))
      EQUIVALENCE (T(1,2),Y(1))
      DATA NOUT,NIN/6,5/
      DATA LABO/4H X +,4H LAG/
      DATA CHAR/1H*,1H+,1HX,1H#,1H./
      DATA XNAME/10*1H ,1HX,9*1H /
      DATA YNAME/10*1H ,1HY,9*1H /
      DATA LABDX/4HORIG,4HINAL,4H DAT,4HA - ,4HX   ,15*4H    /
      DATA LABDY/4HORIG,4HINAL,4H DAT,4HA - ,4HY   ,15*4H    /
      DATA CAPT/4HSCAT,4HTER ,4HPLOT,4H OF ,4HX VS,4H. Y ,14*4H    /
      DATA CRNK/4H- RA,4HNK T,4HRANS,4HFORM,4HATIO,4HN   /
C
      LAGT=0
      WRITE(NOUT,1)
    1 FORMAT(1H1)
      READ(NIN,10) NTAPE,IDOX,IDOY,MORD,IPLT1,IPLT2,IDST,KDEL,IOUTD,
     +IREG,IUNIV
   10 FORMAT(12I5)
      IF(IUNIV.EQ.1) READ(NIN,10) KLAG1,KLAG2
      IF(IDOX.EQ.9) READ(NIN,12) BETAPX
      IF(IDOX.EQ.10) READ(NIN,12) BETAPX,BETAWX
      IF(IDOY.EQ.9) READ(NIN,12) BETAPY
      IF(IDOY.EQ.10) READ(NIN,12) BETAPY,BETAWY
   12 FORMAT(2F10.0)
      WRITE(NOUT,20)
   20 FORMAT(//,10X,20(4H****),/,10X,'*',78X,'*',/,10X,'*   BISAM ',
     +'- BIVARIATE DATA ANALYSIS USING FOURIER EXPANSIONS',19X,'*',
     +/,10X,'*              AND QUANTILE TECHNIQUES',44X,'*',/,10X,'*',
     +78X,'*',/,10X,'*    PROGRAMMER: TERRY J. WOODFIELD',T90,'*',
     +/,10X,'*',78X,'*',/,10X,20(4H****))
      IF(IUNIV.EQ.0) GO TO 28
      CALL DATAIN(NTAPE,W,NW,L,LABX)
      DO 200 KLAG=KLAG1,KLAG2
      N=NW-KLAG
      DO 22 I=1,N
      X(I)=W(I)
      Y(I)=W(I+KLAG)
   22 CONTINUE
      DO 24 I=1,20
   24 LABY(I)=LABX(I)
      CALL ICODEA(KLAG,8H(I4)    ,LABY(17))
      LABY(15)=LABO(1)
      LABY(16)=LABO(2)
      GO TO 40
   28 CONTINUE
      CALL DATAIN(NTAPE,X,NX,L,LABX)
      CALL DATAIN(NTAPE,Y,NY,L,LABY)
      N=NX
      IF(NX.EQ.NY) GO TO 40
      WRITE(NOUT,30) LABX,LABY
   30 FORMAT(1H ,10X,20A4/,10X,20A4,//,10X,'SAMPLE SIZES NOT EQUAL.',
     +' BIVARIATE ANALYSIS INAPROPRIATE. EXECUTION TERMINATED.')
      STOP
   40 IF(LAGT.GE.1) GO TO 51
      WRITE(NOUT,50) LABX,LABY,N,NTAPE,IDOX,IDOY,MORD,IPLT1,IPLT2,
     +IDST,KDEL,IOUTD,IREG,IUNIV
   50 FORMAT(///,10X,20A4,/,10X,20A4,//,10X,'SAMPLE SIZE =',I5,
     +//,10X,'OPTIONS FOR THIS ANALYSIS:',/,T20,'NTAPE = ',I2,
     +T40,'IDOX = ',I3,T60,'IDOY = ',I2,/,T20,'MORD = ',I2,
     +T40,'IPLT1 = ',I2,T60,'IPLT2 = ',I2,/,T20,'IDST = ',I2,
     +T40,'KDEL = ',I3,T60,'IOUTD = ',I3,/,T20,'IREG = ',I3,T40,
     +'IUNIV = ',I2)
      IF((IPLT1.EQ.1).OR.(IPLT1.EQ.3))
     +CALL PPLOT(X,Y,500,N,1,CHAR,CAPT,XNAME,YNAME,0)
   51 WRITE(NOUT,1)
C
C   ORDER BIVARIATE DATA BY X VALUES
C
      CALL ORD2(T,N,500)
      IF((IDST.EQ.0).OR.(LAGT.GE.1)) GO TO 52
      BETAP=BETAPX
      BETAW=BETAWX
```

```
      CALL QUENT(N,X,LABX,LABDX,IDQX,N+4,X25,XMED,X75,XBAR,SDX)
      BETAP=BETAPY
      BETAW=BETAWY
      CALL QUENT(N,Y,LABY,LABDY,IDQY,N+4,X25,YMED,X75,YBAR,SDY)
C
C   TRIM DATA SET OF AT MOST KDEL EXTREME POINTS
C
   52 CALL TRIM(X,Y,XMED,YMED,KDEL,N,NEWN)
      N=NEWN
C
C   OBTAIN RANKS OF X AND Y VALUES
C
      CALL RANK(X,N,RANKX)
      CALL RANK(Y,N,RANKY)
C
C   COMPUTE CORRELATION COEFFICIENTS
C
      CALL SPRMN(N,RHO,SUMD)
      CALL KENDAL(N,TAUA,TAUB,SOMER,NC,ND,NIND,NDEP,NPAIRS)
      CALL PEARSN(N,R)
C
C   OBTAIN ESTIMATES OF BIVARIATE DEPENDENCE DENSITY AND DENSITY
C   QUANTILE FUNCTION
C
      CALL CMPINF(N,MORD,IDQX,IDQY,IPLT2,IOUTD,IREG,LABX,LABY,MD,AIC)
      WRITE(NOUT,1)
      IF((NIND.EQ.0).AND.(NDEP.EQ.0)) GO TO 59
      WRITE(NOUT,55) NIND,NDEP
   55 FORMAT(//,10X,'TIES IN X =',I4,'. TIES IN Y =',I4,//)
   59 WRITE(NOUT,60) LABX,LABY,N
   60 FORMAT(//,10X,20A4,/,10X,20A4,//,10X,'SAMPLE SIZE =',I5)
C
C   OBTAIN INFORMATION MEASURES FOR NORMAL CASE
C
      RI=-0.5*ALOG(1.-R*R)
      RHOI=-0.5*ALOG(1.-RHO*RHO)
      TAUAI=-0.5*ALOG(1.-TAUA*TAUA)
      TAUBI=-0.5*ALOG(1.-TAUB*TAUB)
      SOMERI=-0.5*ALOG(1.-SOMER*SOMER)
C
C   WRITE VALUES OF CORRELATION COEFFICIENTS
C
      WRITE(NOUT,64) R,RI,RHO,RHOI,TAUA,TAUAI,TAUB,TAUBI,SOMER,SOMERI
   64 FORMAT(//,T10,'CORRELATION COEFFICIENT',T45,'VALUE',T55,
     +'INFORMATION (NORMAL CASE)',/,T10,23(1H-),T40,10(1H-),T55,
     +25(1H-),/,T20,'PEARSON',T40,F10.6,T60,F15.6,/,T20,'SPEARMAN',
     +T40,F10.6,T60,F15.6,/,T20,'KENDALL A',T40,F10.6,T60,F15.6,/,
     +T20,'KENDALL B',T40,F10.6,T60,F15.6,/,T20,9HSOMER'S D,T40,
     +F10.6,T60,F15.6)
      DO 66 I=1,4
   66 MD(I)=-MD(I)
      WRITE(NOUT,68) MD(4),AIC(4),MD(1),AIC(1),MD(2),AIC(2),MD(3),
     +AIC(3)
   68 FORMAT(//,T10,'MODEL',T35,'INFORMATION',T67,'AIC',/,T10,8(1H-),
     +T35,11(1H-),T60,10(1H-),/,T10,'D-TILDA',T35,F11.5,T60,F10.5,/,
     +T10,'D-HAT 8',T35,F11.5,T60,F10.5,/,T10,'D-HAT 24',T35,F11.5,
     +T60,F10.5,/,T10,'D-HAT 48',T35,F11.5,T60,F10.5)
      DO 80 I=1,N
      X(I)=RANKX(I)/FLOAT(N+1)
      Y(I)=RANKY(I)/FLOAT(N+1)
   80 CONTINUE
      DO 90 I=1,6
      CAPT(I+8)=CRNK(I)
   90 CONTINUE
      IF((IPLT1.EQ.2).OR.(IPLT1.EQ.3))
     +CALL PPLOT(X,Y,500,N,1,CHAR,CAPT,XNAME,YNAME,0)
C
      LAST=1
      IPLT2=0
  200 CONTINUE
      STOP
      END
      FUNCTION AREST(X,L,OPTKNM,OPTCOE)
C*********************************************************************
C   FUNCTION TO COMPUTE AUTOREGRESSIVE ESTIMATOR EVALUATED AT X
C   METHOD: ARTWO = OPTKNM / ABS(1 + Y)**2
C           WHERE Y = OPTCOE(J)*EXP(I*J*2*PI*X) SUMMED OVER J = 1, L
C   INPUT:
C         X: SCALAR AT WHICH AUTOREGRESSIVE ESTIMATE IS EVALUATED.
C         L: ORDER. MUST BE LESS THAN 11. SEE METHOD.
C         OPTKNM: SEE METHOD.
C         OPTCOE: AUTOREGRESSIVE COEFFICIENTS OF ORDER L. SEE METHOD.
C                 OPTCOE IS A COMPLEX 10-VECTOR.
C   OUTPUT: FUNCTION RETURNS VALUE OF AUTOREGRESSIVE ESTIMATOR EVALUATED
C           AT X.
C   SUBROUTINES CALLED: NONE.
C*********************************************************************
      COMPLEX OPTCOE(L)
      COMPLEX G
      PI=4.*ATAN(1.0)
      G=CMPLX(1.,0.)
      DO 1 J=1,L
      FJ=J
      G=G+OPTCOE(J)*CEXP(CMPLX(0.,X*2.*PI*FJ))
    1 CONTINUE
      AREST=OPTKNM/REAL(G*CONJG(G))
      RETURN
      END
      SUBROUTINE AUTDEN(W,N,IDQW,IPLT2,MORD,ALPH,RVARW,SIGO,NVW,
     +ISORT,WLAB)
C*************************************************************
C
C   THIS SUBPROGRAM COMPUTES A SMOOTHED DENSITY QUANTILE
C   FUNCTION BASED ON THE AUTOREGRESSIVE METHOD OF PARZEN(1979).
C
C   INPUT: W - RAW DATA
C          N - SAMPLE SIZE
C          IDQW - INDICATOR FOR NULL DIST. OF W
C          ISORT - 0 IF W AND RANKW SORTED, 1 OTHERWISE.
C          MORD - MAXIMUM ALLOWABLE ORDER (<=6)
C          IPLT2 - 0--> NO PLOTS
C                  1--> PLOT OF AR DENSITY-QUANTILE FUNCTION
C          WLAB - VARIABLE NAME FOR W IN A4 FORMAT
C
C   OUTPUT: NVW - ORDER OF AUTOREGRESSIVE DENSITY ESTIMATOR
C           ALPH - COEFFICIENTS FOR AUTOREGRESSIVE REPRESENTATION
C           RVARW - RESIDUAL VARIANCE FOR BEST ORDER
C           SIGO - INTEGRATING FACTOR (SIGMA-TILDA FOR NULL MODEL)
C
C   SUBPROGRAMS CALLED: GRD,QTOFQ,WSPACE,FORIER,AUTORG,PARZ,
C                       AREST,FQFNC,MONRIS,QFIND,PLOTXY,FTERP,MINMAX,
C                       MIN,MAX
C
C*************************************************************
      COMMON /PARM/ BETAP,BETAW
      DIMENSION W(N),RVAR(6),U(500),QN(500),QL(500),FQ(500),
     +WXS(500),CWXS(500),ILOC(6),CAT(6),WK1(500)
      DIMENSION CAPT(20)
      COMPLEX A(5),PHI(5),ALPH(5),ALPHA(15),RESVAR
      DATA CAPT/4HUNIV,4HARIA,4HTE D,4HENSI,4HTY-Q,4HUANT,4HILE ,
     +4HFOR ,4HRAND,4HOM V,4HARIA,4HBLE ,6*4H    /
      DATA SPCFAC/0.1/
```

```
      CAPT(13)=WLAB                                                     BISAM
      WRITE(6,1) WLAB                                                   BISAM
    1 FORMAT(//,10X,'UNIVARIATE DENSITY ESTIMATION RESULTS FOR ',       BISAM
     +'VARIABLE ',A4,//)                                                BISAM
      DO 5 I=1,N                                                        BISAM
      QN(I)=W(I)                                                        BISAM
    5 CONTINUE                                                          BISAM
      N2=N+2                                                            BISAM
      M=MORD+1                                                          BISAM
      IF(M.GT.6) M=6                                                    BISAM
      MM1=M-1                                                           BISAM
      H=1./FLOAT(N)                                                     BISAM
      IF(ISORT.EQ.0) GO TO 10                                           BISAM
      CALL QUICK(N,QN)                                                  BISAM
   10 CONTINUE                                                          BISAM
C                                                                       BISAM
C   COMPUTE N EQUALLY SPACED U VALUES BETWEEN 0 AND 1                   BISAM
C                                                                       BISAM
      U(1)=.5*H                                                         BISAM
      DO 30 J=1,N                                                       BISAM
      U(J+1)=U(J)+H                                                     BISAM
   30 CONTINUE                                                          BISAM
C                                                                       BISAM
C   COMPUTE LITTLE Q AND FQ=1/(LITTLE Q)                                BISAM
C                                                                       BISAM
      NP1=N+1                                                           BISAM
      CALL QTOFQ(QN,U,NP1,QL,SPCFAC)                                    BISAM
C                                                                       BISAM
C   COMPUTE WEIGHTED SPACINGS (LITTLE D(U)) BASED ON IDQN DIST.         BISAM
C                                                                       BISAM
      DO 40 I=1,N                                                       BISAM
   40 WK1(I)=FQFNC(U(I+1),IDQN)                                         BISAM
      CALL WSPACE(WXS,CWXS,NP1,QL,WK1,U,SIGO)                           BISAM
C                                                                       BISAM
C   COMPUTE FOURIER TRANSFORM OF WEIGHTED SPACINGS                      BISAM
C                                                                       BISAM
      CALL FORIER(WXS,U(2),N,A,M)                                       BISAM
C                                                                       BISAM
C   COMPUTE AUTOREGRESSIVE COEFFICIENTS FOR ORDERS 1 TO M               BISAM
C                                                                       BISAM
      II=1                                                              BISAM
      DO 100 K=1,MM1                                                    BISAM
      KP1=K+1                                                           BISAM
      CALL AUTORG(A,KP1,M,ALPH,PHI,RESVAR)                              BISAM
      RVAR(K)=REAL(RESVAR)                                              BISAM
      ILOC(K)=II                                                        BISAM
      DO 90 J=1,K                                                       BISAM
      ALPHA(II)=ALPH(J)                                                 BISAM
      II=II+1                                                           BISAM
   90 CONTINUE                                                          BISAM
  100 CONTINUE                                                          BISAM
      CALL PARZ(RVAR,M-1,N,CAT,NVW)                                     BISAM
      IF(NVW.EQ.0) GO TO 115                                            BISAM
      LOC=ILOC(NVW)                                                     BISAM
      DO 110 I=1,NVW                                                    BISAM
      ALPH(I)=ALPHA(LOC)                                                BISAM
      LOC=LOC+1                                                         BISAM
  110 CONTINUE                                                          BISAM
  115 CALL CLPLT1(RVAR,M-1,1,4HRVAR,41,1)                               BISAM
C                                                                       BISAM
C   COMPUTE UNIVARIATE DENSITY-QUANTILE AT 100 POINTS AND PLOT          BISAM
C                                                                       BISAM
      WRITE(6,120) SIGO                                                 BISAM
  120 FORMAT(/,10X,'SIGO = ',F10.4)                                     BISAM
      RVARW=RVAR(NVW)                                                   BISAM
      DO 150 I=1,100                                                    BISAM
      U(I)=FLOAT(I)/101.0                                               BISAM
      FI=1.0                                                            BISAM
      IF(NVW.GT.0) FI=AREST(U(I),NVW,RVARW,ALPH)                        BISAM
      IF(FI.EQ.0.) FI=H                                                 BISAM
      FQ(I)=FQFNC(U(I),IDQN)/(FI*SIGO)                                  BISAM
  150 CONTINUE                                                          BISAM
      IF(IPLT2.EQ.1)                                                    BISAM
     +CALL PLOTXY(U,FQ,100,CAPT,4H   U,4H  FQ,1)                        BISAM
      RETURN                                                            BISAM
      END                                                               BISAM
      SUBROUTINE AUTORG(A,LSP1,M,ALPHA,PHI,FKM)                         BISAM
C*****************************************************************      BISAM
C     COMPUTES THE COEFFICIENTS ALPHA(.) AND FKM OF THE                 BISAM
C     AUTOREGRESSIVE ESTIMATOR ACCORDING TO A RECURSIVE                 BISAM
C     ALGORITHM                                                         BISAM
C     INPUT :                                                           BISAM
C          A : VECTOR OF COMPLEX FOURIER TRANSFORM,                     BISAM
C              OF DIMENSION AT LEAST M                                  BISAM
C          M : (M-1) IS THE MAXIMUM ORDER OF SCHEME                     BISAM
C              TO BE COMPUTED                                           BISAM
C          LSP1 : ORDER OF SCHEME BEING COMPUTED PLUS 1. LSP1.GE.2      BISAM
C     OUTPUT :                                                          BISAM
C          ALPHA : VECTOR OF COEFFICIENTS DEFINING THE                  BISAM
C              APPROXIMATING FUNCTION, HAS TO BE DIMEN-                 BISAM
C              SIONED AT LEAST M AND DECLARED COMPLEX                   BISAM
C          FKM : SCALES THE AUTOREGRESSIVE ESTIMATOR TO                 BISAM
C              INTEGRATE TO A(1), DECLARED REAL                         BISAM
C     ALPHA, PHI AND FKM ARE USED RECURSIVELY,IE. THEIR                 BISAM
C     VALUES AT OUTPUT FOR ORDER J ARE USED AS INPUT                    BISAM
C     FOR ORDER (J+1)                                                   BISAM
C*****************************************************************      BISAM
      COMPLEX A(LSP1),ALPHA(LSP1),PHI(LSP1),G,FJM                       BISAM
      COMPLEX FKM                                                       BISAM
      LS=LSP1-1                                                         BISAM
      TWOPI=8.*ATAN(1.0)                                                BISAM
      FJM=CMPLX(0.,0.)                                                  BISAM
      PHI(LS) = CMPLX(1.,0.)                                            BISAM
      IF(LS .EQ. 1) FKM = CONJG(A(1))                                   BISAM
      DO 4 I = 1,LS                                                     BISAM
    4 FJM=FJM+CONJG(A(I+1)*PHI(I))                                      BISAM
      G=-FJM/FKM                                                        BISAM
      ALPHA(LS) = G                                                     BISAM
      IF(LS .EQ. 1) GO TO 5                                             BISAM
      K = LS -1                                                         BISAM
      DO 2 I = 1,K                                                      BISAM
    2 ALPHA(I) = ALPHA(I) + G*PHI(I)                                    BISAM
    5 CONTINUE                                                          BISAM
      DO 3 I=1,LS                                                       BISAM
    3 PHI(I) = CONJG(ALPHA(LS+1-I))                                     BISAM
      FKM=FKM-FJM*CONJG(FJM)/CONJG(FKM)                                 BISAM
      RETURN                                                            BISAM
      END                                                               BISAM
      SUBROUTINE CLPLT1(X,N,INIT,NAME,MM,IOPT)                    CLPLT1 BISAM
C***************************************************************  CLPLT1 BISAM
C                                                                 CLPLT1 BISAM
C   SUBROUTINE TO PRINT AND PRINTER PLOT THE N-VECTOR X.          CLPLT1 BISAM
C                                                                 CLPLT1 BISAM
C   INPUT :                                                       CLPLT1 BISAM
C           N,X                                                   CLPLT1 BISAM
C           INIT : PRINTED INDEX OF FIRST PRINTED X               CLPLT1 BISAM
C           NAME : 4 CHARACTER LITERAL CONSTANT GIVING            CLPLT1 BISAM
C                  LABEL FOR X                                    CLPLT1 BISAM
C           MM : NUMBER OF COLUMNS IN PLOT (LE.101)               CLPLT1 BISAM
C           IOPT : 1,2 (POINT OR BAR PLOT)                        CLPLT1 BISAM
C                                                                 CLPLT1 BISAM
C   SUBROUTINES CALLED : MAX,MIN                                  CLPLT1 BISAM
```

```
      C                                                                 CLPLT1  BISAM
      C*****************************************************************CLPLT1  BISAM
      C                                                                 CLPLT1  BISAM
            DIMENSION X(N),AL(101)                                      CLPLT1  BISAM
            DATA NIN,NOUT/5,6/                                          CLPLT1  BISAM
            DATA BLANK,DOT,Z/1H ,1H.,1H*/                                       BISAM
      C                                                                 CLPLT1  BISAM
            IOPTX=0                                                     CLPLT1  BISAM
            IF(N.GT.1) GO TO 10                                         CLPLT1  BISAM
            WRITE(NOUT,11) NAME,X(1)                                    CLPLT1  BISAM
         11 FORMAT(10X,A4,'(1) = ',F15.8)                                       BISAM
            GO TO 99                                                    CLPLT1  BISAM
         10 CONTINUE                                                    CLPLT1  BISAM
      C                                                                 CLPLT1  BISAM
      C   INITIALIZE AL :                                               CLPLT1  BISAM
      C                                                                 CLPLT1  BISAM
            ON=(MM-1)/2                                                 CLPLT1  BISAM
            DO 20 J=1,MM                                                CLPLT1  BISAM
         20 AL(J)=DOT                                                   CLPLT1  BISAM
            WRITE(NOUT,25) NAME,(AL(J),J=1,MM)                          CLPLT1  BISAM
         25 FORMAT(/,14X,1HI,6X,A4/10X,15(1H-),2X,101A1)                CLPLT1  BISAM
            DO 30 J=1,MM                                                CLPLT1  BISAM
         30 AL(J)=BLANK                                                 CLPLT1  BISAM
      C                                                                 CLPLT1  BISAM
      C   FIND MAX AND MIN :                                            CLPLT1  BISAM
      C                                                                 CLPLT1  BISAM
            CALL MAX(X,N,XMAX,IND)                                      CLPLT1  BISAM
            CALL MIN(X,N,XMIN,IND)                                      CLPLT1  BISAM
            RX=XMAX-XMIN                                                CLPLT1  BISAM
            IF(RX.LT.1.E-20) IOPTX=1                                    CLPLT1  BISAM
      C                                                                 CLPLT1  BISAM
      C   PLOT :                                                        CLPLT1  BISAM
      C                                                                 CLPLT1  BISAM
            JJ=INIT                                                     CLPLT1  BISAM
            DO 40 J=1,N                                                 CLPLT1  BISAM
            IF(IOPTX.EQ.1) GO TO 36                                     CLPLT1  BISAM
            C1=(X(J)-XMIN)/RX                                           CLPLT1  BISAM
            C1=2.*(C1-.5)                                               CLPLT1  BISAM
            GO TO 37                                                    CLPLT1  BISAM
         36 C1=0.                                                       CLPLT1  BISAM
         37 K=ON*(C1+1.)+1.5                                            CLPLT1  BISAM
            AL(K)=Z                                                     CLPLT1  BISAM
            IF(IOPT.EQ.1) GO TO 35                                      CLPLT1  BISAM
            DO 39 I=1,K                                                 CLPLT1  BISAM
         39 AL(I)=Z                                                     CLPLT1  BISAM
         35 CONTINUE                                                    CLPLT1  BISAM
            WRITE(NOUT,38) JJ,X(J),(AL(I),I=1,MM)                       CLPLT1  BISAM
         38 FORMAT(10X,I5,F10.4,2X,101A1)                               CLPLT1  BISAM
            JJ=JJ+1                                                     CLPLT1  BISAM
            AL(K)=BLANK                                                 CLPLT1  BISAM
            IF(IOPT.EQ.1) GO TO 40                                      CLPLT1  BISAM
            DO 41 I=1,K                                                 CLPLT1  BISAM
         41 AL(I)=BLANK                                                 CLPLT1  BISAM
         40 CONTINUE                                                    CLPLT1  BISAM
      C                                                                 CLPLT1  BISAM
      C                                                                 CLPLT1  BISAM
         99 CONTINUE                                                    CLPLT1  BISAM
            RETURN                                                      CLPLT1  BISAM
            END                                                         CLPLT1  BISAM
            SUBROUTINE CMPINF(N,MORD,IDOX,IDOY,IPLT2,IOUTD,IREG,LABX,LABY,      BISAM
           +HD,AIC)                                                             BISAM
      C*****************************************************************        BISAM
      C                                                                         BISAM
      C   SUBROUTINE TO COMPUTE COVARIANCE MATRIX OF COMPLEX                    BISAM
      C   EXPONENTIAL "SUFFICIENT STATISTICS" TO BE USED IN                     BISAM
      C   SEQUENTIAL REGRESSION ROUTINE TO OBTAIN "REGRESSION"                  BISAM
      C   MODELS FOR ORDERS 1 THROUGH M*M. SUBROUTINE CPTENT IS USED            BISAM
      C   TO OBTAIN COEFFFICIENTS FOR THREE MAXIMUM ENTROPY                     BISAM
      C   ESTIMATES OF THE BIVARIATE DEPENDENCE DENSITY. THEN THE               BISAM
      C   BIVARIATE DENSITY QUANTILE IS FORMED BY TAKING THE PRODUCT            BISAM
      C   OF THE ESTIMATED DEPENDENCE DENSITY AND THE UNIVARIATE                BISAM
      C   AUTOREGRESSIVE ESTIMATORS.                                            BISAM
      C                                                                         BISAM
      C   INPUT: RANKX,RANKY - VECTORS CONTAINING RANKS OF X AND CORANKS        BISAM
      C                        OF Y                                             BISAM
      C          X,Y - BIVARIATE DATA                                           BISAM
      C          N - SAMPLE SIZE                                                BISAM
      C          MORD - MAXIMUM AUTOREGRESSIVE ORDER TO BE USED FOR             BISAM
      C                 UNIVARIATE AR DENSITY ESTIMATION (<=6)                  BISAM
      C          IDOX,IDOY - NULL DISTRIBUTIONS FOR AUTOREGRESSIVE SMOOTHING    BISAM
      C          IPLT2 - 0--> NO AUTOREGRESSIVE DENSITY PLOTS                   BISAM
      C                  1--> BEST ORDER AR DENSITY PLOTS                       BISAM
      C          IDST - 0--> NO UNIVARIATE DESCRIPTIVE STATISTICS               BISAM
      C                 1--> UNIVARIATE DESCRIPTIVE STATISTICS FOR X AND Y      BISAM
      C          IOUTD - 1 IF THE 3 MODELS ESTIMATED ARE TO HAVE VALUES         BISAM
      C                  WRITTEN TO TAPES 1,2, AND 3; 0 O.W.                    BISAM
      C                  FT01001, ETC. DD JCL CARDS MUST BE INCLUDED IF         BISAM
      C                  IOUTD=1.                                               BISAM
      C          IREG - 1 IF QUANTILE REGRESSION PERFORMED, 0 O.W.              BISAM
      C          LABX,LABY - LABELS FOR X AND Y                                 BISAM
      C                                                                         BISAM
      C   OUTPUT: PHI - COVARIANCE MATRIX                                       BISAM
      C           FOX,FOY - UNIVARIATE DENSITY QUANTILE FUNCTIONS               BISAM
      C           DQHAT - BIVARIATE DENSITY QUANTILE FUNCTION                   BISAM
      C           HD - VECTOR OF ENTROPY ESTIMATORS: 1 - ORDER 8 MODEL          BISAM
      C                                              2 - ORDER 24 MODEL         BISAM
      C                                              3 - ORDER 48 MODEL         BISAM
      C                                              4 - RAW (FROM D-TILDA)     BISAM
      C           AIC - VALUES OF ENTROPY CRITERION FUNCTION BASED ON           BISAM
      C                 AKAIKE'S INFORMATION CRITERION                          BISAM
      C                                                                         BISAM
      C   NOTE: FOX,FOY ARE NOT PASSED BACK TO THE CALLING PROGRAM.             BISAM
      C         ALSO, CRITERION FUNCTIONS ARE PLOTTED BUT NOT PASSED            BISAM
      C         BACK TO THE CALLING PROGRAM.                                    BISAM
      C                                                                         BISAM
      C   SUBPROGRAMS CALLED: CSQREG,CPTENT,PLOTXY,FTERP,AUTDEN,MIN             BISAM
      C                                                                         BISAM
      C*****************************************************************        BISAM
      C                                                                         BISAM
            COMMON /DATAR/ X(500),Y(500),RANKX(500),RANKY(500)                  BISAM
            COMMON /PARM/ BETAP,BETAW                                           BISAM
            DIMENSION IND(97),RADSQ(500),IORD(49),                              BISAM
           +HD(4),AIC(4),QTIL(500),AMAT(40,40)                                  BISAM
            DIMENSION MENT(3),PSI(3),LABX(20),LABY(20)                          BISAM
            COMPLEX ARGM(12,12),PHI(50,50),CEXP,CONJG,CMPLX,ZARG                BISAM
            COMPLEX ALPHX(5),ALPHY(5),COF(97)                                   BISAM
            DATA IORD/25,24,18,17,26,32,19,31,33,23,11,16,10,30,12,9,           BISAM
           +27,39,20,38,34,40,13,37,41,22,4,15,3,29,5,8,2,36,6,1,28,            BISAM
           +46,21,45,35,47,14,44,42,48,7,43,49/                                 BISAM
            DATA MENT/8,24,48/                                                  BISAM
            REAL LQDHAT                                                         BISAM
            IF(N.GT.29) GO TO 20                                                BISAM
            WRITE(6,10) N                                                       BISAM
         10 FORMAT(10X,'SAMPLE SIZE ',I2,' IS TOO SMALL. CMPINF SKIPPED.')      BISAM
            RETURN                                                              BISAM
      C                                                                         BISAM
      C   SET VALUES OF CONSTANTS                                               BISAM
      C                                                                         BISAM
         20 N2=N-2                                                              BISAM
            DO 21 I=1,4                                                         BISAM
         21 HD(I)=999.0                                                         BISAM
      C                                                                         BISAM
```

```
C     FOR THIS VERSION USING COMPLEX SEQUENTIAL REGRESSION THE
C     MAXIMUM APPROXIMATING ORDER IS SET AT 7.
C
      M=7
      L=MOD(M,2)
      ML=(M-L)/2
      IF(L.EQ.0) M=M+1
      M2=2*M-1
      MM=M*M
      M1=MM+1
      MM1=MM-1
      DENOM=1.0/FLOAT(N+1)
      TWOPI=8.0*ATAN(1.0)
      PI=TWOPI/2.0
C
C     COMPUTE NEAREST NEIGHBOR DENSITY ESTIMATE AND RAW ESTIMATE
C     OF ENTROPY
C
      HDO=0.0
      DO 30 I=3,N2
      DO 25 J=1,N
      RADSQ(J)=(RANKX(I)-RANKX(J))**2+(RANKY(I)-RANKY(J))**2
   25 CONTINUE
      DO 26 K=1,6
      CALL MIN(RADSQ,N,RMIN,INDR)
      RADSQ(INDR)=FLOAT(2*N*N)
   26 CONTINUE
      VKJ=RMIN*DENOM*DENOM*PI
      IF(VKJ.EQ.0.0) VKJ=0.5*DENOM*DENOM*PI
C
C     DTIL IS ALOG(DTIL)
C
      DTIL(I)=ALOG(5.0/(FLOAT(N+1)*VKJ))
      HDO=HDO-DTIL(I)
   30 CONTINUE
      HD(4)=HDO/FLOAT(N-4)
C
C     COMPUTE MATRIX OF EXPONENTIAL CROSS-PRODUCTS TO BE USED FOR
C     COVARIANCE COMPUTATIONS
C
      DO 50 I=1,M2
      I1=I-M
      DO 50 J=1,M2
      J1=J-M
      ARGM(I,J)=CMPLX(0.0,0.0)
      DO 40 K=3,N2
      ARG=TWOPI*(FLOAT(I1)*RANKX(K)+FLOAT(J1)*RANKY(K))*DENOM
      ZARG=CMPLX(0.,ARG)
      ARGM(I,J)=ARGM(I,J)+CEXP(ZARG)
   40 CONTINUE
      ARGM(I,J)=ARGM(I,J)/FLOAT(N-4)
   50 CONTINUE
C
C     COMPUTE COVARIANCE MATRIX
C
      DO 60 IN=1,MM
      I=IN-1
      I2=MOD(I,M)
      I1=(I-I2)/M+M-ML
      I2=I2+M-ML
      DO 55 JN=1,IN
      J=JN-1
      J2=MOD(J,M)
      J1=(J-J2)/M
      II=I1-J1+ML
      JJ=I2-J2+ML
      J1=J1+M-ML
      J2=J2+M-ML
      PHI(IN,JN)=ARGM(II,JJ)-ARGM(I1,I2)*CONJG(ARGM(J1,J2))
      PHI(JN,IN)=CONJG(PHI(IN,JN))
   55 CONTINUE
   60 CONTINUE
C
C     COMPUTE LAST ROW OF COVARIANCE MATRIX
C
      DBAR=0.0
      DO 70 I=3,N2
      DBAR=DBAR+DTIL(I)
   70 CONTINUE
      DBAR=DBAR/FLOAT(N-4)
      DO 90 IN=1,MM
      I=IN-1
      I2=MOD(I,M)
      I1=(I-I2)/M-ML
      I2=I2-ML
      PHI(M1,IN)=CMPLX(0.0,0.0)
      DO 80 K=3,N2
      ARG=TWOPI*(FLOAT(I1)*RANKX(K)+FLOAT(I2)*RANKY(K))*DENOM
      ZARG=CMPLX(0.0,ARG)
      PHI(M1,IN)=PHI(M1,IN)+DTIL(K)*CONJG(CEXP(ZARG))
   80 CONTINUE
      PHI(M1,IN)=PHI(M1,IN)/FLOAT(N-4)-DBAR*CONJG(ARGM(I1+M,I2+M))
      PHI(IN,M1)=CONJG(PHI(M1,IN))
   90 CONTINUE
      PHI(M1,M1)=0.0
      DO 100 K=3,N2
      PHI(M1,M1)=PHI(M1,M1)+DTIL(K)*DTIL(K)
  100 CONTINUE
      PHI(M1,M1)=PHI(M1,M1)/FLOAT(N-4)-DBAR*DBAR
C
C     CALL ROUTINE CPTENT TO COMPUTE AND PLOT RESIDUAL VARIANCE AND
C     DETERMINE THREE MODELS FOR D(U1,U2)
C
      CALL CPTENT(N,M,PHI,IORD,. 'L,COF,MENT)
C
C     COMPUTE UNIVARIATE DENSITY ESTIMATES USING AUTOREGRESSIVE
C      TECHNIQUE
C
      WRITE(6,134)
  134 FORMAT(1H1)
      CALL AUTDEN(X,N,IDOX,IPLT2,MORD,ALPHX,RVARX,SIGX,NVX,0,4H   X )
      WRITE(6,134)
      CALL AUTDEN(Y,N,IDOY,IPLT2,MORD,ALPHY,RVARY,SIGY,NVY,1,4H   Y )
      WRITE(6,135) NVX,NVY
  135 FORMAT(//,10X,'UNIVARIATE BEST ORDERS: NVX =',I3,', NVY =',I3)
      DO 200 ITER=1,3
      PSI(ITER)=0.0
      ENT=0.0
      WRITE(6,136) MENT(ITER)
  136 FORMAT(1H1,10X,'RESULTS FOR ORDER ',I2,' MODEL:',
     *//,10X,'U1',17X,'U2',15X,'DONT',15X,'DHAT',/,11X,
     *88(1H-))
      DOMAX=0.
      LOCIT=1
      IF(ITER.EQ.2) LOCIT=MENT(1)+1
      IF(ITER.EQ.3) LOCIT=MENT(1)+MENT(2)+1
      DO 220 I=1,40
      U1=(FLOAT(I)-.5)*0.025
      DO 220 J=1,40
      U2=(FLOAT(J)-0.5)*0.025
C
C     COMPUTE VALUES OF UNIVARIATE DENSITY-QUANTILE FUNCTIONS
```

```
C
      FOX=1.0
      IF(NVX.GT.0) FOX=AREST(U1,NVX,RVARX,ALPHX)
      FOX=FOFNC(U1,IDOX)/(FOX*SIGX)
      FOY=1.0
      IF(NVY.GT.0) FOY=AREST(U2,NVY,RVARY,ALPHY)
      FOY=FOFNC(U2,IDOY)/(FOY*SIGY)
C
C   COMPUTE BIVARIATE DENSITY QUANTILE BY FORMING PRODUCT
C   OF DEPENDENCE DENSITY AND AUTOREGRESSIVE ESTIMATORS
C
      LGDHAT=0.0
      KP=MENT(ITER)
      LOC=LOCIT
      DO 200 K=1,KP
      II=IND(LOC)-1
      I2=MOD(II,M)
      I1=(II-I2)/M-ML
      I2=I2-ML
      ARG=TWOPI*(FLOAT(I1)*U1+FLOAT(I2)*U2)
      ZARG=CMPLX(0.0,ARG)
      LGDHAT=LGDHAT+REAL(COF(LOC)*CEXP(ZARG))
      LOC=LOC+1
  200 CONTINUE
      IF(LGDHAT.LE.170.) GO TO 203
      WRITE(6,202) U1,U2,LGDHAT
  202 FORMAT(/,10X,'LOG DHAT(',F6.6,',',F6.6,')=',E20.6,
     *' IN CMPINF. BIVARIATE MODELING TERMINATED.')
      RETURN
  203 IF(LGDHAT.LT.-20.) LGDHAT=-20.
      DHAT=EXP(LGDHAT)
      ENT=ENT-LGDHAT*DHAT
      PSI(ITER)=PSI(ITER)+DHAT
      DONT=DHAT*FOX*FOY
      AMAT(I,J)=DONT
      IF(DONT.LE.DOMAX) GO TO 205
      DOMAX=DONT
      U1MAX=U1
      U2MAX=U2
  205 CONTINUE
      IMOD=MOD(I,8)
      JMOD=MOD(J,8)
      IF((IMOD.EQ.0).AND.(JMOD.EQ.0)) WRITE(6,210) U1,U2,DONT,DHAT
      IF(IOUTD.EQ.1) WRITE(ITER,210) U1,U2,DONT,DHAT
  210 FORMAT(2X,4F15.8)
C
  220 CONTINUE
      PSI(ITER)=PSI(ITER)/1681.0
      ENT=ENT/1681.0
      HD(ITER)=ENT/PSI(ITER)+ALOG(PSI(ITER))
      WRITE(6,224)
  224 FORMAT(//,10X,20(4H****),//)
      WRITE(6,225) MENT(ITER),PSI(ITER)
  225 FORMAT(/,10X,'INTEGRATING FACTOR FOR ORDER ',I3,' IS ',F10.4)
      U1N=U1MAX*FLOAT(N)
      U2N=U2MAX*FLOAT(N)
      DOMAX=DOMAX/PSI(ITER)
      WRITE(6,227) U1MAX,U2MAX,DOMAX,U1N,U2N
  227 FORMAT(/,10X,'MAXIMUM VALUE FOR DEPENDENCE DENSITY QUANTILE:',
     *//,T20,'D(',F7.5,',',F7.5,') = ',F15.7,//,T20,'U1*N = ',
     *F7.2,5X,'U2*N = ',F7.2)
      WRITE(6,230)
  230 FORMAT(//,10X,'COEFFICIENTS FOR BIVARIATE DEPENDENCE DENSITY',
     *//,12X,'NU1',2X,'NU2',2X,' REAL(COF) IMAG(COF)',/,10X,33(1H-))
      LOC=LOCIT
      DO 245 I=1,KP
      II=IND(LOC)-1
      I2=MOD(II,M)
      I1=(II-I2)/M-ML
      I2=I2-ML
      WRITE(6,240) I1,I2,COF(LOC)
  240 FORMAT(10X,2I5,2F10.4)
      LOC=LOC+1
  245 CONTINUE
C
C   DISPLAY CONTOUR PLOT OF DEPENDENCE DENSITY QUANTILE
C
      WRITE(6,255) LABX,LABY,MENT(ITER)
  255 FORMAT(1H1,/,10X,20A4,/,10X,20A4,/,10X,'CONTOUR PLOT FOR ',
     *'BIVARIATE DENSITY QUANTILE - ORDER = ',I2,//)
      CALL CPLOT(AMAT,40,40,40)
      WRITE(6,257)
  257 FORMAT(//,30X,'ORDINATE IS U1, ABSCISSA IS U2',/,30X,
     *'U1 CORRESPONDS TO X (FIRST VARIABLE), U2 TO Y')
  260 CONTINUE
C
C   DETERMINE BEST MODEL BY ENTROPY CRITERION
C
      DO 270 I=1,3
  270 AIC(I)=(HD(I)-2.*FLOAT(MENT(I))/FLOAT(N))-HD(4)
      AIC(4)=0.
      KMOD=1
      DO 280 I=2,3
  280 IF(AIC(I).GE.0.) KMOD=I
      WRITE(6,290) MENT(KMOD)
  290 FORMAT(/,10X,'BEST MODEL BY AIC IS ORDER ',I2,' MODEL.')
      LOCO=1
      IF(KMOD.EQ.2) LOCO=MENT(1)+1
      IF(KMOD.EQ.3) LOCO=MENT(1)+MENT(2)+1
      KPARM=MENT(KMOD)
      PSIO=PSI(KMOD)
C
C   PERFORM QUANTILE REGRESSION
C
      IF(IREG.EQ.1) CALL QREG(N,M,ML,PSIO,LOCO,KPARM,COF,IND,RADSQ,
     *DTIL)
      RETURN
      END
      SUBROUTINE CPLOT(A,IA,M,N)
C****************************************************************
C
C   SUBROUTINE TO PROVIDE A 10 TIERED CONTOUR PLOT OF THE MATRIX
C   A. THIS ROUTINE IS DESIGNED SPECIFICALLY TO PROVIDE A
C   CONTOUR PLOT OF THE DEPENDENCE DENSITY QUANTILE FROM PROGRAM
C   BISAM.
C
C   INPUT: A - MATRIX TO BE PLOTTED
C          M,N - ROW AND COLUMN DIMENSIONS OF A
C          IA - ROW DIMENSION OF A IN CALLING PROGRAM
C
C   SUBPROGRAMS CALLED: MINMAX
C
C   PROGRAMMER: PHIL SPECTOR
C   MODIFIED BY TERRY J. WOODFIELD
C
C****************************************************************
      DIMENSION A(IA,N),XNUM(11)
      DIMENSION ISYMB(8),LINE1(50),LINE2(50),LINE3(50),LINEN(110)
      DATA ISYMB/1H.,1H',1H-,1H=,1H+,1HO,1HX,1HA/
      DATA IBLANK/1H /
      DIVN=1./FLOAT(N)
      DIVM=1./FLOAT(M)
```

```
      CALL MINMAX(A,IA,M,N,XMIN,XMAX)
      FACT=(XMAX-XMIN)/18.
      DO 80 I=1,M
      IFLAG=0
      IFLAG1=0
      DO 10 K=1,80
      LINE1(K)=IBLANK
      LINE2(K)=IBLANK
   10 LINE3(K)=IBLANK
      DO 70 J=1,N
      ITEMP=IFIX(((A(I,J)-XMIN)/FACT)+.5)+1
      INEXT=(ITEMP+2)/2
      IF(ITEMP.EQ.1)GO TO 11
      IF(ITEMP.GE.18)GO TO 18
      IF((ITEMP.EQ.16).OR.(ITEMP.EQ.17))GO TO 17
      IF((ITEMP.EQ.14).OR.(ITEMP.EQ.15))GO TO 15
      LINE1(J)=ISYMB(INEXT)
      GO TO 70
   11 LINE1(J)=ISYMB(1)
      GO TO 70
   18 IFLAG=1
      IFLAG1=1
      LINE1(J)=ISYMB(6)
      LINE2(J)=ISYMB(7)
      LINE3(J)=ISYMB(8)
      GO TO 70
   17 IFLAG=1
      LINE1(J)=ISYMB(6)
      LINE2(J)=ISYMB(7)
      GO TO 70
   15 IFLAG=1
      LINE1(J)=ISYMB(6)
      LINE2(J)=ISYMB(3)
   70 CONTINUE
      U=(FLOAT(I)-0.5)*DIVM
      WRITE(6,901)U,LINE1
      IF(IFLAG.EQ.1)WRITE(6,902)LINE2
      IF(IFLAG1.EQ.1)WRITE(6,902)LINE3
   80 CONTINUE
      INDEX=(N+5)/5
      DO 81 I=1,INDEX
   81 XNUM(I)=(5.*FLOAT(I)-0.5)*DIVN
      NEND=INDEX*10
      DO 82 I=1,NEND
   82 LINEN(I)=ISYMB(3)
      DO 83 I=11,NEND,10
   83 LINEN(I)=ISYMB(5)
      NEND=NEND+1
      IF(NEND.GE.110)GO TO 85
      DO 84 I=NEND,110
   84 LINEN(I)=IBLANK
   85 WRITE(6,903)LINEN
      WRITE(6,904)(XNUM(I),I=1,INDEX)
      RETURN
  901 FORMAT(1H ,2X,F5.3,2X,'|',1X,80(A1,1X))
  902 FORMAT(1H+,11X,80(A1,1X))
  903 FORMAT(1H ,9X,110A1)
  904 FORMAT(1H ,12X,11(5X,F5.3))
      END
      SUBROUTINE CPTENT(N,M,PHI,IORD,IND,COF,MENT)
C*****************************************************************
C
C   SUBPROGRAM TO COMPUTE MAXIMUM ENTROPY ESTIMATES OF
C   THE BIVARIATE DEPENDENT DENSITY.
C
C   THREE FITTED MODELS WILL BE RETURNED WITH COEFFICIENTS
C   IN COF, VARIABLE NAMES (INDICES) IN IND, THE FIRST LOCATION
C   OF COEFFICIENTS AND INDICES FOR THE 2ND MODEL IN MENT(1)+1, I.E.,
C   COF(MENT(1)+1) CONTAINS COEFFICIENT NUMBER ONE OF THE SECOND
C   REGRESSION MODEL CORRESPONDING TO INDEX IND(MENT(1)+1), ETC.
C
C   INPUT: N,M - SAMPLE SIZE, UNIVARIATE MAXIMUM ORDER (M**2
C                USED FOR BIVARIATE MAX ORDER)
C          PHI - COVARIANCE MATRIX
C          IORD - VECTOR OF ORDERED INDICES FOR SEQUENTIAL REGRESSION
C
C   AUXILLIARY: NVAR,RVAR,BEST - VECTORS AND MATRIX
C               FROM ROUTINE CSOREG
C
C   OUTPUT: COF,IND - SEE ABOVE
C
C   SUBROUTINES CALLED: CSOREG,CSWEEP,CLPLT1
C
C*****************************************************************
C
      DIMENSION MENT(3),IND(97)
      DIMENSION IORD(49),NVAR(49),INDV(1225),RVAR(49)
      COMPLEX PHI(50,50),COF(97),BEST(1225)
      MM=M*M
      MM1=MM-1
      L=MOD(M,2)
      ML=(M-L)/2
      TWOPI=8.0*ATAN(1.0)
C
C   CALL ROUTINE CSOREG TO PERFORM SEQUENTIAL REGRESSION ON PHI
C
      CALL CSOREG(PHI,50,MM,IORD,BEST,INDV,1225,RVAR,NVAR,NIVN)
      CALL CLPLT1(RVAR,NIVN,1,4HRVAR,41,1)
C
C   PLACE COEFFICIENTS IN COF FOR EACH ORDER
C
C
      LOC=1
      DO 180 I=1,3
      K=MENT(I)
      IF(K.EQ.0) GO TO 180
      K1=NVAR(K)
      DO 170 KK=1,K
      IND(LOC)=INDV(K1)
      COF(LOC)=BEST(K1)
      K1=K1+1
      LOC=LOC+1
  170 CONTINUE
  180 CONTINUE
      RETURN
      END
      SUBROUTINE CSOREG(A,NDIM,NIV,IORD,BEST,INDV,MDIM,RVAR,NVAR,NIVN)
C*****************************************************************
C
C   SUBPROGRAM TO PERFORM SEQUENTIAL REGRESSION USING COVARIANCE
C   OR CORRELATION MATRIX A(NIV+1,NIV+1).
C
C   INPUT: A - COVARIANCE MATRIX (COMPLEX)
C          NDIM - ROW DIMENSION OF A IN CALLING PROGRAM
C          NIV - NUMBER OF INDEPENDENT VARIABLES
C          IORD - INTEGER VECTOR CONTAINING INDICES OF VARIABLES
C                 IN THE ORDER THEY ARE TO BE ENTERED INTO THE MODEL
C          MDIM - DIMENSION OF BEST IN CALLING PROGRAM
C
C   OUTPUT: A - SWEPT COVARIANCE MATRIX
C           BEST,INDV - VECTORS OF SUBSET INFORMATION
C                 BEST CONTAINS LEAST SQUARES PARAMETER ESTIMATES
```

```
C                 INDV CONTAINS VARIABLE INDICES                          BISAM
C             RVAR - VECTOR OF RESIDUAL VARIANCES                         BISAM
C             IORD - VECTOR CONTAINING INDICES OF VARIABLES IN ORDER      BISAM
C                    THAT THEY WERE ENTERED WITH VALUES CAUSING           BISAM
C                    SINGULARITIES IN A OMITTED                           BISAM
C             NIVN - NUMBER OF INDEPENDENT VARIABLES INCLUDED IN          BISAM
C                    ANALYSIS                                             BISAM
C                                                                         BISAM
C   SUBPROGRAMS CALLED: CSWEEP                                            BISAM
C                                                                         BISAM
C*****************************************************************        BISAM
C                                                                         BISAM
      COMPLEX A(NDIM,NDIM),BEST(NDIM)                                     BISAM
      DIMENSION INDV(NDIM),IORD(NIV),RVAR(NIV),NVAR(NIV)                  BISAM
      DATA TOL/1.E-20/                                                    BISAM
      NV=NIV+1                                                            BISAM
      NIVN=NIV                                                            BISAM
      VAR=REAL(A(NV,NV))                                                  BISAM
      LOC=1                                                               BISAM
      LC2=1                                                               BISAM
      KOUNT=1                                                             BISAM
      K=1                                                                 BISAM
   10 ID=IORD(K)                                                          BISAM
      KOUNT=KOUNT+1                                                       BISAM
      TEST=REAL(A(ID,ID))**2+AIMAG(A(ID,ID))**2                           BISAM
      IF(TEST.LE.TOL) GO TO 40                                            BISAM
      CALL CSWEEP(A,NDIM,NV,ID,ID)                                        BISAM
      RVAR(K)=REAL(A(NV,NV))/VAR                                          BISAM
      DO 30 KK=1,K                                                        BISAM
      KID=IORD(KK)                                                        BISAM
      IF(KK.NE.1) GO TO 20                                                BISAM
      NVAR(LC2)=LOC                                                       BISAM
      LC2=LC2+1                                                           BISAM
   20 INDV(LOC)=KID                                                       BISAM
      BEST(LOC)=-A(NV,KID)                                                BISAM
      LOC=LOC+1                                                           BISAM
   30 CONTINUE                                                            BISAM
      GO TO 60                                                            BISAM
   40 NIVN=NIVN-1                                                         BISAM
      DO 50 I=K,NIVN                                                      BISAM
      IORD(I)=IORD(I+1)                                                   BISAM
   50 CONTINUE                                                            BISAM
      GO TO 10                                                            BISAM
   60 K=K+1                                                               BISAM
      IF(KOUNT.LE.NIV) GO TO 10                                           BISAM
      RETURN                                                              BISAM
      END                                                                 BISAM
      SUBROUTINE CSWEEP(A,NDIM,N,K1,K2)                                   BISAM
C*********************************************************                BISAM
C                                                                         BISAM
C   SUBROUTINE TO SWEEP THE NXN COMPLEX MATRIX A ON ITS K1                BISAM
C   THRU K2 DIAGONAL ELEMENTS (SWP(K)SWP(K)A=A)                           BISAM
C                                                                         BISAM
C   INPUT :                                                               BISAM
C           A,N,K1,K2                                                     BISAM
C           NDIM : ROW DIMENSION OF A IN CALLING PROGRAM                  BISAM
C                                                                         BISAM
C   OUTPUT :                                                              BISAM
C           A                                                             BISAM
C                                                                         BISAM
C   SUBROUTINES CALLED : NONE                                             BISAM
C                                                                         BISAM
C**********************************************************               BISAM
C                                                                         BISAM
      COMPLEX D,A(NDIM,NDIM)                                              BISAM
      DATA NOUT/6/                                                        BISAM
C                                                                         BISAM
C   FIX DIAGONAL K :                                                      BISAM
C                                                                         BISAM
      DO 50 K=K1,K2                                                       BISAM
C                                                                         BISAM
C   CHECK FOR ZERO :                                                      BISAM
C                                                                         BISAM
      TEST=REAL(A(K,K))**2+AIMAG(A(K,K))**2                               BISAM
      IF(TEST.LT.1.E-25) GO TO 99                                         BISAM
      D=1./A(K,K)                                                         BISAM
      A(K,K)=1.                                                           BISAM
C                                                                         BISAM
C   KTH ROW :                                                             BISAM
C                                                                         BISAM
      DO 10 I=1,N                                                         BISAM
   10 A(K,I)=D*A(K,I)                                                     BISAM
C                                                                         BISAM
C   KTH COLUMN :                                                          BISAM
C                                                                         BISAM
         DO 20 J=1,N                                                      BISAM
         IF(J.EQ.K) GO TO 20                                              BISAM
         A(J,K)=-A(J,K)*D                                                 BISAM
   20    CONTINUE                                                         BISAM
C                                                                         BISAM
C   OTHERS :                                                              BISAM
C                                                                         BISAM
         DO 40 J=1,N                                                      BISAM
         IF(J.EQ.K) GO TO 40                                              BISAM
            DO 30 I=1,N                                                   BISAM
            IF(I.EQ.K) GO TO 30                                           BISAM
            A(J,I)=A(J,I)+A(J,K)*A(K,I)/D                                 BISAM
   30       CONTINUE                                                      BISAM
   40    CONTINUE                                                         BISAM
C                                                                         BISAM
C                                                                         BISAM
   50 CONTINUE                                                            BISAM
C                                                                         BISAM
C                                                                         BISAM
      GO TO 110                                                           BISAM
   99 WRITE(NOUT,100) K,K1,K2                                             BISAM
  100 FORMAT(10X,I2,15HTH DIAG OF FROM,1X,I2,1X,2HTO,1X,                  BISAM
     1I2,1X,17HIS ZERO IN CSWEEP)                                         BISAM
  110 RETURN                                                              BISAM
      END                                                                 BISAM
      SUBROUTINE DATAIN(NTAPE,X,N,L,LAB)                          DATAIN  BISAM
C**********************************************************       DATAIN  BISAM
C   SUBROUTINE TO READ A DATA FILE FROM TAPE NTAPE AS FOLLOWS :   DATAIN  BISAM
C                                                                 DATAIN  BISAM
C     CARD1 : LAB1(1),....,LAB(20)  (20A4)                         DATAIN BISAM
C     CARD2 : SAMPLE SIZE N, FORMAT L(1),....,L(5),(I5,4X,5A4)    DATAIN  BISAM
C     CARD3,CARD4,... : DATA X(1),....,X(N) IN L FORMAT           DATAIN  BISAM
C                                                                 DATAIN  BISAM
C**********************************************************       DATAIN  BISAM
C                                                                 DATAIN  BISAM
      DIMENSION X(1),L(5),LAB(20)                                  DATAIN BISAM
C                                                                 DATAIN  BISAM
      READ(NTAPE,1) LAB                                           DATAIN  BISAM
    1 FORMAT(20A4)                                                DATAIN  BISAM
      READ(NTAPE,2) N,L                                           DATAIN  BISAM
    2 FORMAT(I5,4X,5A4)                                                   BISAM
      READ(NTAPE,L) (X(I),I=1,N)                                  DATAIN  BISAM
C                                                                 DATAIN  BISAM
      RETURN                                                      DATAIN  BISAM
      END                                                         DATAIN  BISAM
      SUBROUTINE DESTAT(X,N,NAME,IHEAD,L,INIT,IOUT,Q25,Q50,Q75,XBAR,SD)   BISAM
C********************************************************************     BISAM
```

```
C     SUBROUTINE TO PRINT ORDERED ARRAY BY QUANTILES AND COMPUTE
C     DESCRIPTIVE STATISTICS.
C     INPUT:
C           X: ARRAY OF ORDER STATISTICS
C           N: DIMENSION OF ARRAY X
C           NAME: NAME OF DATA SET.MUST BE ARRAY OF DIMENSION 20 IN
C                 CALLING PROGRAM.
C           IHEAD: HEADING FOR ANALYSIS
C           IUNIT: NUMBER OF UNIT OUTPUT IS DESIRED ON.
C           INIT: 0 FOR FIRST CALL, 1 THEREAFTER
C           IOUT: 1 IF QUANTILES TO BE LISTED, 0 OTHERWISE
C     OUTPUT: PRINTED OUTPUT IS ON IUNIT.
C     NO SUBROUTINES CALLED.
C*********************************************************************
        DIMENSION X(N),NAME(20),SUM(4),SUMSQ(4)
        DIMENSION ALF(3),L(4)
        DIMENSION IHEAD(20)
        DATA ALF/.05,.10,.25/
        NUNIT = 6
C     COMPUTE L,THE ARRAY OF QUARTILE SIZES
        IF(INIT .EQ. 0) GOTO 5
        IF(L(4) .EQ. N) GOTO 25
5       LL = N/4
        L1 = LL
        L2 = LL
        L3 = LL
        L4 = LL
        IBRAN = MOD(N,4) + 1
        GOTO (20,11,12,13),IBRAN
     11 CONTINUE
        L4 = LL + 1
        GO TO 20
     12 CONTINUE
        L1 = LL + 1
        L4 = LL + 1
        GO TO 20
     13 CONTINUE
        L2 = LL + 1
        L3 = LL + 1
        L4 = LL + 1
     20 CONTINUE
        L(1)=L1
        L(2)=L(1) + L2
        L(3)=L(2) + L3
        L(4)=L(3) + L4
C     PRINT DATA ARRAY - ONE COLUMN FOR EACH QUARTER.
25      WRITE(NUNIT,1001) NAME
        WRITE(NUNIT,1020) IHEAD
        IF(IOUT.EQ.0) GOTO 35
        WRITE(NUNIT,1002)
        WRITE(NUNIT,1003)
        DO 30 I = 1,LL
     30 WRITE(NUNIT,1004) I,X(I),X(L(1) + I),X(L(2)+I),X(L(3) + I)
        WRITE(NUNIT,1005)
        IF(L1 .GT. LL) WRITE(NUNIT,1006) X(L(1))
        IF(L2 .GT. LL) WRITE(NUNIT,1007) X(L(2))
        IF(L3 .GT. LL) WRITE(NUNIT,1008) X(L(3))
        IF(L4 .GT. LL) WRITE(NUNIT,1009) L4,X(L(4))
   35   IF(INIT .EQ. 1) RETURN
C     COMPUTE AND PRINT DESCRIPTIVE STATISTICS.
        K = 1
        S = 0.
        SSQ = 0.
        DO 50 I = 1,4
        SI = 0.0
        SQI = 0.0
        KK = L(I)
        DO 40 J = K,KK
        SI = SI + X(J)
        SQI = SQI + X(J)*X(J)
     40 CONTINUE
        K = 1 + L(I)
        S = S + SI
        SSQ = SSQ + SQI
        SUM(I) = SI
        SUMSQ(I) = SQI
     50 CONTINUE
        IF(IOUT.EQ.0) GO TO 55
        WRITE(NUNIT,1010) (SUM   (I),I=1,4)
        WRITE(NUNIT,1011) (SUMSQ (I),I=1,4)
   55   XBAR = S/FLOAT(N)
        VAR = (SSQ -S*XBAR)/FLOAT(N-1)
        SD = SQRT(VAR)
        Q25 = QFIND(X,N,.25)
        Q50 = QFIND(X,N,.50)
        Q75 = QFIND(X,N,.75)
        QIQ = Q75 - Q25
        XBARIQ = (XBAR - Q50) / (2. * QIQ)
        SDIQ   = SD / (2. * QIQ)
        SDIQLG = ALOG(SDIQ)
        SSQN=SSQ/FLOAT(N)
        TRIMN = (Q25+2.*Q50+Q75)*.25
        GASTY = .3*QFIND(X,N,1./3.)+.4*Q50+.3*QFIND(X,N,2./3.)
        WRITE(NUNIT,1012)
        WRITE(NUNIT,1013) N,Q25,Q50,Q75,QIQ,TRIMN,GASTY
        WRITE(NUNIT,1014) SSQN,XBAR,VAR,SD,XBARIQ,SDIQ,SDIQLG
        WRITE(NUNIT,1016)
        DO 60 I = 1,3
        IG = INT(ALF(I)*FLOAT(N))
        M = N -2*IG
        NMGM1 = N - IG - 1
        IGP2 = IG + 2
        TRM = X(IG+1)+X(N-IG)
        WMN = X(IG+1)*FLOAT(IG) + X(N-IG)*FLOAT(IG)
        DO 70 J = IGP2,NMGM1
        TRM = TRM + X(J)
        WMN = WMN + X(J)
     70 CONTINUE
        TRM = TRM/M
        WMN = WMN/FLOAT(N)
        WRITE(NUNIT,1017) ALF(I),WMN,TRM
60      CONTINUE
    999 CONTINUE
        RETURN
   1001 FORMAT(///T20,20A4)
   1002 FORMAT(/T40,'ORDER STATISTICS IN QUARTERS'/T40,28(1H*))
   1003 FORMAT(T20,' SEQUENCE'/T20,'   WITHIN'/T20,' QUARTILE   ',
       +'FIRST QUARTER  SECOND QUARTER   THIRD QUARTER  FOURTH QUARTER'
       + /T21,8(1H*),2(3X,13(1H*),2X,14(1H*)))
   1004 FORMAT(T20,I6 , 4(1X,F15.4))
   1005 FORMAT(1X)
   1006 FORMAT(1H+,T29,F15.4)
   1007 FORMAT(1H+,T45,F15.4)
   1008 FORMAT(1H+,T61,F15.4)
   1009 FORMAT(1H+,T20,I6,T77,F15.4)
   1010 FORMAT(1H0,T20,'SUM',5X,4(1X,F15.4))
   1011 FORMAT(1H0,T20,'SUM OF'/T20,' SQUARES',1X,4(1X,F15.4))
  1012  FORMAT(///T45,'DESCRIPTIVE  STATISTICS'/T45,23(1H*))
  1013  FORMAT(//' SUMSTAT',T20,'SAMPLE',T28,'   LOWER',T52,'   UPPER  ',
       +        T64,'INT_QUARTL',T88,'GASTWIRTHS'/' SUMSTAT',T20,' SIZE'
       +        ,T28,' QUARTILE',T40,'  MEDIAN',T52,' QUARTILE',
       +        T64,'  RANGE',T76,' TRIMEAN',T88,' ESTIMATE'
```

```
     *           ,//' SUMSTAT',T20,I5,T29,6(G11.4,1X))
 1016 FORMAT(///' SUMSTAT',T20,'SUMSQ/N',' MEAN',T40,' VARIANCE',T52,
     *       ' STD DEV',T65,'MEAN 10',T76,'STD DEV 10',T88,
     *       'LOG STD 10'//' SUMSTAT',T16,7(G11.4,1X))
 1015 FORMAT(///T40,'TRUNCATION',T52,'WINSORIZED',T64,'  TRIMMED'/
     *       T40,'   POINT',T52,'   MEAN',T64,'   MEAN'/)
 1017 FORMAT(T40,F7.3,T52,2(G11.4,1X))
 1030 FORMAT(T20,20A6)
      END
      SUBROUTINE FCODEA(X,IFORM,NAME)
C***************************************************
C   SUBROUTINE TO CONVERT REAL VARIABLE X
C   WHICH HAS 4 CHARACTER F-FORMAT IFORM
C   TO 8 CHARACTER ALPHAMERIC ARRAY NAME WHICH IS
C   IN A FORMAT.
C   INPUT  : NSCRCH : SCRATCH TAPE NUMBER
C            X, IFORM
C   OUTPUT : NAME(1),NAME(2) : 4 CHARACTERS EACH
C***************************************************
      DIMENSION NAME(2),IFORM(2)
      NSCRCH = 11
      REWIND NSCRCH
      WRITE(NSCRCH,IFORM)X
      REWIND NSCRCH
      READ(NSCRCH,10)NAME
   10 FORMAT(2A4)
      RETURN
      END
      SUBROUTINE FORIER(F,U,N,A,MA)
C*****************************************************************
C   SUBROUTINE TO COMPUTE THE FOURIER TRANSFORM
C   PHI(V) OF A DENSITY DEFINED ON (0,1) FOR V=0,1,...,M
C   INPUT :
C          F,U,N : VECTORS OF LENGTH N CONTAINING F(U),U
C          MA : MAXIMUM VALUE OF V FOR WHICH PHI(V) IS COMPUTED
C   OUTPUT : A : COMPLEX-VALUED VECTOR CONTAINING THE PHI'S
C   SUBROUTINES CALLED : NONE
C******************************************************************
      DIMENSION F(N),U(N)
      COMPLEX A(MA),Z
      TWOPI=8.*ATAN(1.)
      FN=FLOAT(N)
      DO 20 IM=1,MA
      FIM=IM-1
      A(IM)=CMPLX(0.,0.)
      DO 10 I=1,N
         Z=CMPLX(0.,TWOPI*FIM*U(I))
   10    A(IM)=A(IM)+F(I)*CEXP(Z)
      A(IM)=A(IM)/FLOAT(N)
   20 CONTINUE
      A(1)=CMPLX(1.,0.)
      RETURN
      END
      FUNCTION FOFNC(X,IDON)
C**********************************************************
C  ROUTINE TO COMPUTE THE VARIOUS DENSITY-QUANTILE FUNCTIONS
C     INPUT:
C          X - VALUE AT WHICH THE FUNCTION IS TO BE COMPUTED
C          IDON - INDICATOR FOR THE DESIRED FUNCTION,
C                 MUST BE IN THE EXCLUSIVE RANGE 1-11
C**********************************************************
      COMMON /PARM/ BETAP,BETAW
      GO TO(1,2,3,4,5,6,7,8,9,10,11),IDON
C  COMPUTE THE NORMAL
 1    CONTINUE
      IF( X .LT. .001) GO TO 101
      IF( X .GT. .999) GO TO 101
      PI=4.*ATAN(1.0)
      CALL MDNRIS(X,T,IER)
      FOOO=EXP(-0.5*T**2)/(SQRT(2.*PI))
      FOFNC = FOOO
      GO TO 99
  101 FOOO=0.
      FOFNC = FOOO
      GO TO 99
C  COMPUTE THE EXPONENTIAL
 2    CONTINUE
      FOOO=1.-X
      FOFNC = FOOO
      GO TO 99
C  COMPUTE THE LOGISTIC
 3    CONTINUE
      FOFNC = X*(1. - X)
      GO TO 99
C  COMPUTE THE DOUBLE EXPONENTIAL
 4    CONTINUE
      IF(X .LE. .5) FOFNC = X
      IF(X .GT. .5) FOFNC = 1. - X
      GO TO 99
C  COMPUTE THE UNIFORM RECIPROCAL
 5    CONTINUE
      FOFNC = (1. - X)**2
      GO TO 99
C  COMPUTE THE CAUCHY
 6    CONTINUE
      PI = 4.*ATAN(1.0)
      FOFNC = SIN(PI*X)**2/PI
      GO TO 99
C  COMPUTE THE EXTREME VALUE
 7    CONTINUE
      IF(X .EQ. 1.) GO TO 102
      FOFNC = (X - 1.)*ALOG(1. - X)
      GO TO 99
  102 CONTINUE
      FOFNC = 0.
      GO TO 99
C  COMPUTE THE LOG NORMAL
 8    CONTINUE
      IF(X .EQ. 1.) GO TO 103
      IF(X .EQ. 0.) GO TO 103
      CALL MDNRIS(X,T,IER)
      PI = 4.*ATAN(1.0)
      FOFNC = (-0.5*T**2-T)/SQRT(2.*PI)
      FOFNC = EXP(FOFNC)
      GO TO 99
  103 FOFNC = 0.0
      GO TO 99
C  COMPUTE THE PARETO
 9    CONTINUE
      FOFNC = (1.-X)**(1.+BETAP)/BETAP
      GO TO 99
C  COMPUTE THE WEIBULL
 10   CONTINUE
      IF(X .EQ. 1.) GO TO 104
      FOFNC = (1.-X)*(-ALOG(1.-X))**(1.-BETAW)/BETAW
      GO TO 99
  104 CONTINUE
      FOFNC = 0.
      GO TO 99
C  COMPUTE THE HALF LOGISTIC
 11   CONTINUE
      FOFNC=1.-X**2
```

```
   90  RETURN
       END
       SUBROUTINE FTERP(U,V,X,F,N,M)
C***************************************************************
C
C   SUBROUTINE TO PERFORM LINEAR INTERPOLATION ON V TO
C   OBTAIN F AT THE M X VALUES
C
C   INPUT: U - VECTOR OF VALUES AT WHICH V EVALUATED
C          V - FUNCTION VALUES TO INTERPOLATE
C          X - VALUES AT WHICH INTERPOLATED FUNCTION TO BE
C              EVALUATED
C          N - DIMENSION OF VECTORS U AND V
C          M - DIMENSION OF VECTORS X AND F
C
C   NOTE: ALL ABSCISSA VECTORS MUST BE ORDERED
C
C   OUTPUT: F - INTERPOLATED FUNCTION VALUES
C
C***************************************************************
       DIMENSION U(N),V(N),X(M),F(M)
       IF(N.EQ.M) GO TO 100
       II=1
       DO 60 I=1,M
   10  IF(X(I)-U(II))20,40,50
   20  IF(II.NE.1) GO TO 30
       F(I)=V(1)+(V(2)-V(1))*(X(I)-U(1))/(U(2)-U(1))
       GO TO 60
   30  F(I)=V(II-1)+(V(II)-V(II-1))*(X(I)-U(II-1))/(U(II)-U(II-1))
       GO TO 60
   40  F(I)=V(II)
       GO TO 60
   50  II=II+1
       IF(II.LT.N) GO TO 10
       II=N
       GO TO 30
   60  CONTINUE
  100  RETURN
       END
       SUBROUTINE ICODEA(K,IFORM,NAME)
C*****************************************************
C   SUBROUTINE TO CONVERT INTEGER VARIABLE K
C   WHICH HAS 8 CHARACTER I-FORMAT IFORM
C   TO 8 CHARACTER ALPHAMERIC ARRAY NAME WHICH IS
C   IN A-FORMAT.
C   INPUT  : NSCRCH : SCRATCH TAPE NUMBER
C            K, IFORM
C   OUTPUT : NAME(1),NAME(2) : 4 CHARACTERS EACH
C*******************************************************
       DIMENSION NAME(2),IFORM(2)
       NSCRCH=11
       REWIND NSCRCH
       WRITE(NSCRCH,IFORM)K
       REWIND NSCRCH
       READ(NSCRCH,10)NAME
    10 FORMAT(2A4)
       RETURN
       END
       SUBROUTINE KENDAL(N,TAUA,TAUB,SOMER,NC,ND,NIND,NDEP,NPAIRS)
C
C*****************************************************************
C
C   SUBROUTINE TO COMPUTE KENDALL'S TAU-A AND TAU-B (FOR TIED
C   RANKS), AND SOMER'S D.
C
C   INPUT: RANKX - THE VECTOR OF RANKS OF THE INDEPENDENT VARIABLE
C          RANKY - THE VECTOR OF RANKS OF THE DEPENDENT VARIABLE
C          N - NUMBER OF PAIRED OBSERVATIONS.
C
C   OUTPUT: TAUA - KENDALL'S TAU-A ASSUMING NO TIES.
C           TAUB - KENDALL'S TAU-B FOR TIED RANKS.
C           SOMER - SOMER'S D FOR TIED RANKS IN THE DEPENDENT
C                   VARIABLE.
C           NC - NUMBER OF CONCORDANT PAIRS
C           ND - NUMBER OF DISCORDANT PAIRS
C           NIND - NUMBER OF PAIRS WITH TIES IN THE INDEPENDENT
C                  VARIABLE BUT NOT IN THE DEPENDENT VARIABLE
C           NDEP - NUMBER OF PAIRS WITH TIES IN THE DEPENDENT
C                  VARIABLE BUT NOT IN THE INDEPENDENT VARIABLE
C           NPAIRS - NUMBER OF PAIRS OF BIVARIATE OBSERVATIONS
C
C   SUBROUTINES CALLED: ORD2
C
C*****************************************************************
C
       COMMON /DATAR/ X(500),Y(500),RANKX(500),RANKY(500)
       DIMENSION T1(500,2)
       DO 10 I=1,N
       T1(I,1)=RANKX(I)
       T1(I,2)=RANKY(I)
   10  CONTINUE
C
C   INITIALIZE NC,ND,NIND,NDEP AND BEGIN COUNTING PROCEDURE
C
       NC=0
       ND=0
       NIND=0
       NDEP=0
       I=1
    20 K=I+1
    30 IF(K.GT.N) GO TO 60
C
C   CHECK IF RANKS ARE EQUAL
C
C   TOLERANCE FOR REAL ARITHMETIC: ASSUME X=0 IF ABS(X).LE.5.E-6
C
       TEST1=ABS(T1(I,1)-T1(K,1))
       IF(TEST1.GT.5.E-6) GO TO 40
       TEST2=ABS(T1(I,2)-T1(K,2))
       IF(TEST2.GT.5.E-6) NIND=NIND+1
       K=K+1
       GO TO 30
C
C   COUNT NUMBER OF CONCORDANT AND DISCORDANT PAIRS
C
    40 DO 50 J=K,N
       IF(T1(J,2)-T1(I,2).GT.5.E-6) NC=NC+1
       IF(T1(J,2)-T1(I,2).LT.-5.E-6) ND=ND+1
    50 CONTINUE
    60 I=I+1
       IF(I.LT.N) GO TO 20
C
C   SWITCH COLUMNS OF T1
C
       DO 70 I=1,N
       T1(I,2)=RANKX(I)
       T1(I,1)=RANKY(I)
    70 CONTINUE
C
C   ORDER BY Y VALUES AND COMPUTE NDEP
C
       CALL ORD2(T1,N,500)
```

```
      I=1
   80 K=I+1
   90 IF(K.GT.N) GO TO 100
      TEST1=ABS(T1(I,1)-T1(K,1))
      IF(TEST1.GT.5.E-6) GO TO 100
      TEST2=ABS(T1(I,2)-T1(K,2))
      IF(TEST2.GT.5.E-6) NDEP=NDEP+1
      K=K+1
      GO TO 90
  100 I=I+1
      IF(I.LT.N) GO TO 80
C
C   COMPUTE NONPARAMETRIC CORRELATION COEFFICIENTS BASED ON
C   NUMBER OF CONCORDANT AND DISCORDANT PAIRS
C
      XNUM=FLOAT(NC-ND)
      NPAIRS=N*(N-1)/2
      ADENOM=FLOAT(NPAIRS)
      BDENOM=SQRT(FLOAT(NC+ND+NDEP)*FLOAT(NC+ND+NIND))
      SDENOM=FLOAT(NC+ND+NDEP)
      TAUA=XNUM/ADENOM
      TAUB=XNUM/BDENOM
      SOMER=XNUM/SDENOM
      RETURN
      END
      SUBROUTINE KSD(D,U,N,DM,UM,DP,UP)
C*****************************************************
C   SUBROUTINE TO COMPUTE KOMOLGOROV-SMIRNOFF STATISTIC FOR
C   THE DEVIATIONS D(U)-U.  UPPER AND LOWER BOUNDS ARE GIVEN.
C   INPUT : D,U,N
C   OUTPUT :
C         DP,UP : MAX (+) DEVIATION, DP, WHICH IS AT U=UP
C         DM,UM : MAX (-) DEVIATION, DM, WHICH IS AT U=UM
C   SUBROUTINES CALLED : NONE
C*****************************************************
      DIMENSION D(N),U(N)
      SQN = SQRT(FLOAT(N))
      DP = (D(1) - U(1))
      UP = U(1)
      DM = DP
      UM = UP
      DO 10   I = 2,N
      DT = (D(I) - U(I) )
      IF(DT .LE. DP) GOTO 1
      UP = U(I)
      DP = DT
      GOTO 10
1     IF(DT .GE. DM) GOTO 10
      UM = U(I)
      DM = DT
   10 CONTINUE
      DP = SQN*DP
      DM = SQN*DM
      RETURN
      END
      SUBROUTINE MAX(X,N,XMAX,IND)
C*******************************************************
C
C   SUBROUTINE TO FIND THE MAXIMUM VALUE (XMAX) AND THE
C   INDEX OF THE MAXIMUM VALUE (IND) OF A VECTOR X OF
C   LENGTH N.
C
C   INPUT :
C           N,X(1),...,X(N)
C
C   OUTPUT :
C           XMAX,IND
C
C   SUBROUTINES CALLED : NONE
C
C*******************************************************
C
      DIMENSION X(N)
C
      XMAX=X(1)
      IND=1
      IF(N.EQ.1) RETURN
C
      DO 1 I=2,N
      IF(XMAX.LT.X(I)) IND=I
    1 XMAX=X(IND)
C
      RETURN
      END
      SUBROUTINE MORRIS (P,Y,IER)
C   QUANTILE FUNCTION FOR N(0,1)
      DIMENSION A(65),C(17),D(25),E(23)
      EQUIVALENCE (A(24),C(1)),(A(41),D(1)),  (A(1),E(1) )
      DATA E/.992885376618940,.120467516143104,
     +      .016078199342100,.002686704437162,
     +      .000499634730236,.000098898218599,
     +      .000020391812764,.000004327271618,
     +      .000000938081413,.000000206734721,
     +      .000000046159699,.000000010418680,
     +      .000000002371501,.000000000543026,
     +      .000000000125049,.000000000029138,
     +      .000000000006795,.000000000001591,
     +      .000000000000374,.000000000000088,
     +      .000000000000021,.000000000000005,
     +      .000000000000001/
      DATA C/.912158803417554,-.016266281867664,
     +      .000433556472949,.000214438570074,
     +      .000002625751076,-.000003021091050,
     +      -.000000012406062,.000000062406609,
     +      -.000000000540125,-.000000001423208,
     +      .000000000034384,.000000000033885,
     +      -.000000000001458,-.000000000000810,
     +      .000000000000053,.000000000000020,
     +      -.000000000000002/
      DATA D/.956679709020493,-.023107004309065,
     +      -.004374236097508,-.000576503422651,
     +      -.000010961022307,.000025108547025,
     +      .000010562336068,.000002754412330,
     +      .000000432484498,-.000000020530337,
     +      -.000000043891537,-.000000017684010,
     +      -.000000003991289,-.000000000186932,
     +      .000000000272923,.000000000132817,
     +      .000000000031834,.000000000001670,
     +      -.000000000002036,-.000000000000965,
     +      -.000000000000220,-.000000000000010,
     +      .000000000000013,.000000000000006,
     +      .000000000000001/
      DATA H1,H2,H3,H4,H5/-1.548813042373,
     +      2.565490123147816,-.559457631329832,
     +      2.287915716263624,-1.414213562373095/
      XINF=2**31-1
      X = 1.-P-P
      IER = 0
      SIGMA = SIGN(1.,X)
      IF( .NOT.(X.GT.-1..AND.X.LT.1.)) GO TO 40
      Z = ABS(X)
      IF(Z .GT. .8) GO TO 25
```

```
      W = Z*Z/.32-1.
      N = 22
      IPP = 1
      L = 1
   10 LB2 = 1
      X3 = 1.
      X4 = W
      X6 = A(IPP)
   15 X6 = X6 + A(IPP+LB2) * X4
      X5 = X4 * W * 2.-X3
      X3 = X4
      X4 = X5
      LB2 = LB2 + 1
      IF(LB2 .LE. N) GO TO 15
      GO TO (20,35),L
   20 Y = Z * X6 * SIGMA
      GO TO 40
   25 Z = SQRT(-ALOG(1.-Z*Z))
      IF(Z .GT. .9975) GO TO 30
      W = H1*Z+H2
      IPP = 24
      L = 2
      N = 16
      GO TO 10
   30 W = H3 * Z + H4
      IPP = 41
      N = 24
      L = 2
      GO TO 10
   35 Y = Z * X6 * SIGMA
   40 Y = R2*Y
      RETURN
   45 Y = SIGMA*XINF
      IER = 129
      WRITE(6,101)
  101 FORMAT(//////////1X,35H*****ERROR CONDITION IN MDNRIS*****  ///)
      RETURN
      END
      SUBROUTINE MIN(X,N,XMIN,IND)
C**************************************************
C
C   SUBROUTINE TO FIND THE MINIMUM VALUE (XMIN) AND THE
C   INDEX OF THE MINIMUM VALUE (IND) OF A VECTOR X OF
C   LENGTH N.
C
C   INPUT :
C           N,X(1),....,X(N)
C
C   OUTPUT :
C           XMIN,IND
C
C   SUBROUTINES CALLED : NONE
C
C**************************************************
C
      DIMENSION X(N)
C
      XMIN=X(1)
      IND=1
      IF(N.EQ.1) RETURN
C
      DO 1 I=2,N
      IF(XMIN.GT.X(I)) IND=I
    1 XMIN=X(IND)
C
      RETURN
      END
      SUBROUTINE MINMAX(A,IA,M,N,XMIN,XMAX)
C***************************************************************
C
C   ROUTINE TO FIND THE MINIMUM AND MAXIMUM VALUES OF A MATRIX
C
C   INPUT: N,M,A(N,M),IA=ROW DIMENSION OF A IN CALLING PROGRAM
C
C   OUTPUT: XMIN,XMAX
C
C   SUBROUTINES CALLED: NONE
C
C***************************************************************
C
      DIMENSION A(IA,N)
      XMIN=A(1,1)
      XMAX=A(1,1)
      DO 10 I=1,M
      DO 10 J=1,N
      IF(A(I,J).GT.XMAX)XMAX=A(I,J)
      IF(A(I,J).LT.XMIN)XMIN=A(I,J)
   10 CONTINUE
      RETURN
      END
      SUBROUTINE ORD2(X,N,NDIM)
C
C**********************************************
C
C   SUBROUTINE TO SORT THE NX2-MATRIX X FROM SMALLEST TO
C   LARGEST VALUE OF THE FIRST COLUMN. X IS SORTED IN PLACE.
C
C   THE SORTING ALGORITHM IS DUE TO D. L. SHELL (COMM. ACM,
C   JULY 1959), AND IS FOUND IN KIRCH, A.(1973)
C   'INTRO. TO STATISTICS WITH FORTRAN',NEW YORK:
C   HOLT,RINEHART, AND WINSTON. P. 347.
C
C**********************************************
C
      DIMENSION X(NDIM,2)
      L=N
    1 L=L/2
      IF(L)2,2,3
    2 RETURN
    3 M=N-L
      DO 6 I=1,M
      J=I
    4 JJ=J+L
      IF(X(J,1)-X(JJ,1))6,6,5
C
C   REPLACE ABOVE BY
C
C     IF(X(J,1)-X(JJ,1))5,6,6
C
C   TO ORDER COLUMN 1 IN DESCENDING ORDER
C
    5 X1=X(J,1)
      X2=X(J,2)
      X(J,1)=X(JJ,1)
      X(J,2)=X(JJ,2)
      X(JJ,1)=X1
      X(JJ,2)=X2
      J=J-L
      IF(J)6,6,4
    6 CONTINUE
      GO TO 1
      END
```

```
      SUBROUTINE PARZ(RVAR,M,N,CAT,NORD)
C************************************************************
C   SUBROUTINE TO DETERMINE THE ORDER OF AN AUTOREGRESSIVE
C   PROCESS BY PARZENS CAT CRITERIA
C   INPUT :
C           M,RVAR(1),...,RVAR(M) : STANDARDIZED RES VAR
C           FOR ORDERS 1 THRU M.
C           N : SAMPLE SIZE
C   OUTPUT :
C           NORD : DETERMINED ORDER
C           CAT(1),...,CAT(M)
C   SUBROUTINES CALLED : MIN
C************************************************************
      DIMENSION RVAR(M),CAT(M)
      ON=FLOAT(N)
      DO 1 I=1,M
         C=0.
         DO 2 J=1,I
    2    C=C+((1.-(FLOAT(J)/ON))/RVAR(J))
         C=C/ON
    1 CAT(I)=C-((1.-(FLOAT(I)/ON))/RVAR(I))
      CALL MIN(CAT,M,CATMI,NORD)
      IF(CATMI.GT.-1.-3./ON) NORD=0
      RETURN
      END
      SUBROUTINE PEARSN(N,R)
C
C*****************************************************************
C
C   SUBROUTINE TO COMPUTE PEARSON'S PRODUCT MOMENT CORRELATION
C   COEFFICIENT.
C
C   INPUT: X - THE VECTOR OF VALUES OF THE INDEPENDENT VARIABLE
C          Y - THE VECTOR OF VALUES OF THE DEPENDENT VARIABLE
C          N - THE NUMBER OF BIVARIATE OBSERVATIONS
C
C   OUTPUT: R - PEARSON'S PRODUCT MOMENT CORRELATION COEFFICIENT
C
C*****************************************************************
C
      COMMON /DATAR/ X(500),Y(500),RANKX(500),RANKY(500)
C
C   INITIALIZE VALUES
C
      SUMX=0.
      SUMY=0.
      SUMXY=0.
      SUMXX=0.
      SUMYY=0.
C
C   COMPUTE SUMS OF SQUARES AND CROSS PRODUCTS
C
      DO 10 I=1,N
      SUMX=SUMX+X(I)
      SUMY=SUMY+Y(I)
      SUMXY=SUMXY+X(I)*Y(I)
      SUMXX=SUMXX+X(I)*X(I)
      SUMYY=SUMYY+Y(I)*Y(I)
   10 CONTINUE
C
C   COMPUTE R
C
      SXY=SUMXY-SUMX*SUMY/FLOAT(N)
      SXX=SUMXX-SUMX*SUMX/FLOAT(N)
      SYY=SUMYY-SUMY*SUMY/FLOAT(N)
      R=SXY/SQRT(SXX*SYY)
      RETURN
      END
      SUBROUTINE PLINE(NR,NC,NL,STRT,END,CHARL,PARRAY,VMIN,VMAX,VINC)
      DIMENSION STRT(NL),END(NL),CHARL(NL),PARRAY(NR,NC)
      NCM1 = NC - 1
      FNC = 1. / FLOAT(NCM1)
      NRP1 = NR + 1
C
      DO 10 IL = 1,NL
      IF ( STRT(IL) - END(IL) ) 20,30,40
 20   STRT(IL) = AMAX1(STRT(IL),VMIN)
      END(IL)  = AMIN1(END(IL) ,VMAX)
      GOTO 50
 40   STRT(IL) = AMIN1(STRT(IL),VMAX)
      END(IL)  = AMAX1(END(IL) ,VMIN)
 50   DS = (END(IL) - STRT(IL)) * FNC * VINC
      S  = (STRT(IL) - VMIN   ) * VINC + 1.5
      IND = INT(S)
      DO 60 I = 2,NCM1
      S = S + DS
      IF ( INT(S) .EQ. IND ) GOTO 60
      IND = INT(S)
      PARRAY(NRP1-IND,I) = CHARL(IL)
 60   CONTINUE
      GOTO 10
 30   IF ( STRT(IL) .LE. VMIN .OR. STRT(IL) .GE. VMAX) GOTO 10
      IND = (STRT(IL) - VMIN) * VINC + 1.5
      DO 70 I = 2,NCM1
      PARRAY(NRP1-IND,I) = CHARL(IL)
 70   CONTINUE
 10   CONTINUE
      RETURN
      END
      SUBROUTINE PLOTXY(X,Y,N,CAPT,NAMX,NAMY,IOPT)
C***************************************************************
C
C   SUBROUTINE TO PRINT AND PRINTER PLOT THE N-VECTOR Y AS A
C   FUNCTION OF X.
C
C   INPUT : N,X,Y - X IS ORDERED ON INPUT AND Y(I)=Y(X(I))
C           CAPT - LITERAL CONSTANT FOR TITLE OF PLOT IN 20A4 FORMAT
C           NAMX,NAMY : 4 CHARACTER LITERAL CONSTANTS GIVING
C                       LABELS FOR X AND Y
C           IOPT : 1,2 (POINT OR BAR PLOT)
C
C   SUBROUTINES CALLED : FTERP,MAX,MIN
C
C***************************************************************
C
      DIMENSION X(N),Y(N),T(46),YI(46),CAPT(20),AL(101)
      DATA NOUT/6/
      DATA BLANK,DOT,Z,SL,PLUS/1H ,1H.,1H*,1H|,1H+/
C
      NN=51
      IOPTY=0
      IF(N.GT.10) GO TO 11
      WRITE(NOUT,10) N
   10 FORMAT(10X,'SAMPLE SIZE OF ',I2,' IS TOO SMALL TO PERFORM ',
     +'INTERPOLATION IN PLOTXY.')
      GO TO 100
   11 CONTINUE
      WRITE(NOUT,12) CAPT
   12 FORMAT(1H1,33X,20A4,/)
C
C   CREATE T VECTOR OF EQUALLY SPACED X AND INTERPOLATE TO OBTAIN
C   CORRESPONDING Y VALUES
```

```
C
      DEC=(X(N)-X(1))/45.0
      DO 15 I=1,45
      T(I)=X(1)+FLOAT(I-1)*DEC
   15 CONTINUE
      CALL PTERP(X,Y,T,YI,N,45)
C
C   INITIALIZE AL :
C
      ON=(MM-1)/2
      DO 20 J=1,MM
   20 AL(J)=DOT
      WRITE(NOUT,25) NAMX,NAMY,(AL(J),J=1,MM)
   25 FORMAT(/,15X,A4,6X,A4/10X,20(1H-),2X,101A1)
      DO 30 J=1,MM
   30 AL(J)=BLANK
      AL(1)=BL
      AL(MM)=BL
C
C   FIND MAX AND MIN :
C
      CALL MAX(YI,45,YMAX,IND)
      CALL MIN(YI,45,YMIN,IND)
      RY=1.2*(YMAX-YMIN)
      IF(RY.LT.1.E-20) IOPTY=1
C
C   PLOT :
C
      DO 40 J=1,45
      IF(IOPTY.EQ.1) GO TO 36
      C1=(YI(J)-YMIN)/RY
      C1=2.*(C1-.5)
      GO TO 37
   36 C1=0.
   37 K=ON*(C1+1.)+2.5
      AL(K)=Z
      IF(IOPT.EQ.1) GO TO 35
      DO 39 I=1,K
   39 AL(I)=Z
   35 CONTINUE
      WRITE(NOUT,38) T(J),YI(J),(AL(I),I=1,MM)
   38 FORMAT(10X,F10.4,1X,F9.4,2X,101A1)
      AL(K)=BLANK
      IF(IOPT.EQ.1) GO TO 40
      DO 41 I=2,K
   41 AL(I)=BLANK
   40 CONTINUE
      DO 50 I=1,MM
   50 AL(I)=DOT
      AL(1)=PLUS
      AL(MM)=PLUS
      WRITE(NOUT,60) (AL(I),I=1,MM)
   60 FORMAT(10X,20(1H-),2X,101A1)
C
C
      YMAX=RY+YMIN
      WRITE(NOUT,70) YMIN,YMAX
   70 FORMAT(27X,F10.4,70X,F10.4)
  100 CONTINUE
      RETURN
      END
      SUBROUTINE PPLOT(X,Y,IY,N,NFUN,CHAR,CAPT1,XNAME,YNAME,IOPT)
C*********************************************************************
C
C   SUBROUTINE PPLOT PLOTS UP TO 5 FUNCTIONS ON THE SAME AXIS USING
C   A DIFFERENT SYMBOL FOR EACH.
C
C   INPUT: X - VECTOR OF X VALUES, LENGTH=N
C          Y - VECTOR OR MATRIX OF Y VALUES, SIZE=N BY NFUN
C          IY - ROWS ALLOCATED TO Y IN CALLING PROGRAM
C          N - NUMBER OF X-VALUES
C          NFUN - NUMBER OF FUNCTIONS TO BE PLOTTED (<=5)
C          CHAR - VECTOR OF LENGTH 5 CONTAINING CHARACTERS TO BE
C                 USED IN THE PLOT. (CHAR(1) IS USED FOR THE FIRST
C                 FUNCTION, CHAR(2) FOR THE SECOND, ETC.)
C          CAPT1 - VECTOR OF LENGTH 20 TO BE USED FOR
C                 CAPTION ABOVE THE PLOT (20A4 FORMAT)
C          XNAME,YNAME - 2 VECTORS OF LENGTH 20 TO BE USED FOR
C                 LABELS ON THE X AND Y AXES (20A1 FORMAT)
C          IOPT - IOPT=0 --> SMALLEST POINT WILL BE DIRECTLY ON AXIS
C                 IOPT=1 --> SMALLEST POINT WILL BE SLIGHTLY AWAY
C                 FROM THE AXIS
C
C   SUBROUTINES CALLED: MINMAX
C
C   PROGRAMMER: PHIL SPECTOR
C
C*********************************************************************
      DIMENSION PARRAY(56,76),XJ(16),YJ(16),INDY(5),CHAR(5),CAPT1(20),
     &XNAME(20),YNAME(20),YNAME1(56),X(N),Y(IY,NFUN)
      DATA CAPI,PLUS,DASH,XM,BLANK/1HI,1H+,1H-,1HM,1H /
      DO 1 I=1,19
    1 YNAME1(I)=BLANK
      DO 2 I=1,20
    2 YNAME1(19+I)=YNAME(I)
      DO 3 I=1,18
    3 YNAME1(38+I)=BLANK
      LH=76
      LV=56
      LVP1=LV+1
      LVM1=LV-1
      CALL MINMAX(Y,IY,N,NFUN,YMIN,YMAX)
      CALL MINMAX(X,N,N,1,HMIN,HMAX)
      IF(IOPT.EQ.1)YMIN=YMIN-1.1*((YMAX-YMIN)/FLOAT(LV-1))
      IF(IOPT.EQ.1)HMIN=HMIN-1.1*((HMAX-HMIN)/FLOAT(LH-1))
      VINC=(YMAX-YMIN)/FLOAT(LV-1)
      HINC=(HMAX-HMIN)/FLOAT(LH-1)
      IF(VINC.EQ.0.0)GO TO 99
      DO 10 I=1,LV
      PARRAY(I,1)=CAPI
      DO 5 J=2,LH
      PARRAY(I,J)=BLANK
    5 CONTINUE
   10 CONTINUE
      PARRAY(1,1)=PLUS
      DO 11 I=1,LH
   11 PARRAY(1,I)=DASH
      J=0
      DO 12 I=1,LH,5
      J=J+1
      XJ(J)=HMIN+FLOAT(J)*5.*HINC
   12 PARRAY(1,I)=PLUS
      J=0
      DO 13 I=1,LV,5
      J=J+1
      YJ(J)=YMIN+FLOAT(J)*5.*VINC
   13 PARRAY(I,1)=PLUS
      DO 45 K=1,N
      INDX=(X(K)-HMIN)/HINC+1.0
      DO 44 L=1,NFUN
   44 INDY(L)=(Y(K,L)-YMIN)/VINC+1.0
      PARRAY(INDY(1),INDX)=CHAR(1)
```

```
      IF(NFUN.EQ.1)GO TO 45
      DO 46 I=2,NFUN
      IF(PARRAY(INDY(I),INDX).NE.BLANK)GO TO 49
      PARRAY(INDY(I),INDX)=CHAR(I)
      GO TO 46
   49 PARRAY(INDY(I),INDX)=XM
   46 CONTINUE
   45 CONTINUE
      WRITE(6,902)CAPT1
      K=4
      KOUNT=1
      DO 55 I=1,LVM1
      K=K+1
      IF(K.EQ.5)GO TO 50
      WRITE(6,905)YNAME1(I),(PARRAY(LVP1-I,J),J=1,LN)
      GO TO 55
   50 WRITE(6,906)YNAME1(I),YJ(12-KOUNT),(PARRAY(LVP1-I,J),J=1,LN)
      KOUNT=KOUNT+1
      K=0
   55 CONTINUE
      WRITE(6,906)BLANK,VMIN,(PARRAY(1,J),J=1,LN)
      WRITE(6,915)HMIN,(XJ(I),I=2,16,2)
      WRITE(6,916)(XJ(I),I=1,15,2)
      WRITE(6,920)XNAME
      GO TO 100
   99 WRITE(6,901)
  100 CONTINUE
  901 FORMAT(1H ,'ERROR IN PPLOT: RANGE OF Y VALUES TOO SMALL')
  902 FORMAT(1H1,20X,20A4)
  905 FORMAT(2X,A1,15X,100A1)
  906 FORMAT(2X,A1,3X,F10.3,2X,100A1)
  915 FORMAT(12X,13(F9.4,1X))
  916 FORMAT(18X,8(F9.4,1X))
  920 FORMAT(48X,20A1)
      RETURN
      END
      FUNCTION QFIND(Q,N,QUANT)
C*******************************************************
C     PURPOSE:  TO FIND THE PERCENTILE VALUE OF Q
C               AT QUANT.
C     INPUT:
C           Q - VECTOR OF SIZE N
C           N - NUMBER OF VALUES IN Y
C           QUANT - QUANTILE VALUE
C*******************************************************
      DIMENSION Q(N)
      P = FLOAT(N+1) * QUANT
      I = INT(P)
      P = P-I
      IF(I .NE. 0) GOTO 1
      QFIND = Q(1)
      RETURN
1     IF(I .LT. N) GOTO 2
      QFIND = Q(N)
      RETURN
2     QFIND = (1.-P) * Q(I) + P * Q(I+1)
      RETURN
      END
      SUBROUTINE QPLOT(X,Y,NY,XCHAR,XZ,Z,NZ,IZ,ZCHAR,IORD,VMN,VMX,
     &                 NLINE,STRT,END,CHARL,LX,LY,LC,LI)
C*********************************************************
C     ROUTINE TO DISPLAY PRINTER QUANTILE-BOX PLOT
C     INPUT :
C           X - VECTOR CONTAINING THE VALUES J/(NY+1) WHERE
C               J=1,2,3,...,NY
C           Y - VECTOR OF SIZE NY TO BE PLOTTED
C           XCHAR - CHARACTER FOR Y IN PLOT
C           LX,LY - VECTORS OF SIZE 2 CONTAINING THE LABELS
C                   FOR X AND Y RESPECTIVELY
C           LC - VECTOR OF SIZE 20 CONTAINING THE CAPTION FOR
C                THE PLOT
C           Z - OPTIONAL VECTOR OF SIZE NZ TO BE PLOTTED
C           XZ - ABSCISSA FOR Z
C           NLINE,STRT,END - VALUES FOR SUB. PLINE WHEN Z IS A LINE
C           VMN,VMX - MIN AND MAX VALUES FOR ORDINATE OF PLOT
C           IBOX - EQUAL ZERO IF BOX PLOTS ARE NOT WANTED
C           IZ - EQUAL 1 IF VECTOR Z IS TO BE PLOTTED
C           ZCHAR - PLOTTING CHARACTER TO BE USED FOR THE
C                   Z VECTOR. MUST BE DIFFERENT FROM D.
C           LI - OPTIONAL LABEL TO FOLLOW CAPTION
C           IORD = 1  IF Y IS ORDERED FROM MIN TO MAX
C                = 0  IF Y IS NOT ORDERED
C                = 2 FOR HORIZONTAL ZERO-LINE (FOR IQ PLOT)   .
C*********************************************************
      DIMENSION PARRAY(51,61),CHAR(10),X(NY),Y(NY),LX(2),LY(2),
     &    LC(20),XJ(11),Z(NZ),LI(20),XZ(NZ),STRT(1),END(1),CHARL(1)
      DATA BLANK,DASH,PLUS,CAPI,ZERO / ' ','-','+','|','O' /
      DATA CHAR1,CHAR/'D','O','2','3','4','5','6','7','8','9','M'/
      DATA EPS / 1.E-10 /
      JINC = 5
      KINC = 6
      NR = 1 + 10 * JINC
      NC = 1 + 10 * KINC
      NRP1 = NR + 1
      NRM1 = NR - 1
      NCM1 = NC - 1
      FC = FLOAT(NCM1)
      NUNIT = 6
      CHAR(1) = XCHAR
      IF ( CHAR(1) .EQ. BLANK ) CHAR(1) = CHAR1
      IF(IORD-1) 2,3,4
 2    CALL MIN(Y,NY,VMIN,IER)
      CALL MAX(Y,NY,VMAX,IER)
      IF(IZ .EQ. 0 ) GO TO 1
      CALL MIN(Z,NZ,ZMIN,IER)
      CALL MAX(Z,NZ,ZMAX,IER)
      IF(ZMIN.LT.VMIN)VMIN=ZMIN
      IF(ZMAX.GT.VMAX)VMAX=ZMAX
      GOTO 1
3     VMIN = Y(1)
      VMAX = Y(NY)
      IF(IZ .EQ. 0) GOTO 1
      ZMIN = Z(1)
      ZMAX = Z(NZ)
      IF(ZMIN .LT. VMIN) VMIN = ZMIN
      IF(ZMAX .GT. VMAX) VMAX = ZMAX
      GOTO 1
 4    VMIN = VMN
      VMAX = VMX
 1    CONTINUE
      RANGE = VMAX - VMIN
      IF ( RANGE .LT. EPS ) GOTO 99
      VINC = FLOAT(NRM1) / RANGE
      DO 10 I= 2,NRM1
      PARRAY(I,1)=CAPI
      PARRAY(I,NC) = CAPI
      DO 5 J=2,NCM1
      PARRAY(I,J)=BLANK
 5    CONTINUE
 10   CONTINUE
      DO 20 J=1,NC
      PARRAY(1,J) = DASH
```

```
      PARRAY(NR,J)=DASH
   20 CONTINUE
   15 J=0
      S=0.0
      DO 25 I=1,NC,KINC
      J=J+1
      XJ(J) = S
      S = S + .1
      PARRAY(NR,I)=PLUS
   25 CONTINUE
      DO 30 I = 1,NR,JINC
      PARRAY(NRP1-I,1) = PLUS
   30 CONTINUE
      IF ( NLINE .GE. 1 ) CALL PLINE(NR,NC,NLINE,STRT,END,CHARL,
     &                                 PARRAY,VMIN,VMAX,VINC)
   60 CONTINUE
      DO 45 K=1,NY
      INDX=X(K)*FC + 1.5
      INDY=(Y(K)-VMIN)*VINC + 1.5
      IF(IZ.NE.1)GO TO 70
      INDZ=(Z(K)-VMIN)*VINC + 1.5
      PARRAY(NRP1-INDY,INDX)=CHAR(1)
      PARRAY(NRP1-INDZ,INDX)=ZCHAR
      IF(INDZ.EQ.INDY)PARRAY(NRP1-INDY,INDX)=CHAR(10)
      GO TO 45
   70 CONTINUE
      IF ( CHAR(1) .EQ. XCHAR ) GOTO 44
      DO 42 I=1,10
      IF(PARRAY(NRP1-INDY,INDX).EQ.CHAR(I))GO TO 43
   42 CONTINUE
   44 PARRAY(NRP1-INDY,INDX)=CHAR(1)
      GO TO 45
   43 IF(I.EQ.10)I=I-1
      PARRAY(NRP1-INDY,INDX)=CHAR(I+1)
   45 CONTINUE
      IF ( IZ .NE. 2 ) GOTO 47
      DO 47 I = 1,NZ
      INDXZ = XZ(I) * FC + 1.5
      INDZ  = (Z(I) - VMIN) * VINC + 1.5
      IF ( PARRAY(NRP1-INDZ,INDXZ) .EQ. CHAR(1) ) GOTO 48
      PARRAY(NRP1-INDZ,INDXZ) = ZCHAR
      GOTO 47
   48 PARRAY(NRP1-INDZ,INDXZ) = CHAR(10)
   47 CONTINUE
      WRITE(NUNIT,100)LC,LI
      DYX = (FLOAT(JINC) / FLOAT(NRM1)) * RANGE
      YXALUE = VMAX + DYX
      DO 300 I=1,NR,JINC
      IF(PARRAY(NRP1-I,1).NE.PLUS) PARRAY(NRP1-I,1)=ZERO
  300 CONTINUE
      DO 55 I=1,NRM1
      IF((PARRAY(I,1).EQ.PLUS).OR.(PARRAY(I,1).EQ.ZERO)) GO TO 50
      WRITE(NUNIT,105)(PARRAY(I,J),J=1,NC)
      GO TO 55
   50 YXALUE = YXALUE - DYX
      WRITE(NUNIT,110)YXALUE,(PARRAY(I,J),J=1,NC)
   55 CONTINUE
      WRITE(NUNIT,110)VMIN,(PARRAY(NR,J),J=1,NC)
      WRITE(NUNIT,115)XJ
      WRITE(NUNIT,120)LX,LY
      GO TO 999
   99 CONTINUE
      WRITE(NUNIT,199)LC
      WRITE(NUNIT,200)
  200 FORMAT(' X VALUES')
      WRITE(NUNIT,205)(X(I),I=1,NY)
  205 FORMAT(1X,10F10.5)
      WRITE(NUNIT,210)
  210 FORMAT(' Y VALUES')
      WRITE(NUNIT,205)(Y(I),I=1,NY)
  199 FORMAT(' ERROR IN OPLOT FOR THE PLOT OF ',20A4)
  999 RETURN
  100 FORMAT(1H1,15X,20A4,/,16X,20A4,/)
  105 FORMAT(13X,61A1)
  110 FORMAT(1X,F10.3,2X,61A1)
  115 FORMAT(12X,11(F3.1,3X))
  120 FORMAT(15X,'ABSCISSA IS ',2A4,4X,' , ORDINATE IS ',2A4)
      END
      SUBROUTINE QREG(N,M,ML,PSIQ,LOCO,KPARM,COF,IND,RQ,E)
C*****************************************************************
C
C    SUBROUTINE TO PERFORM QUANTILE REGRESSION
C
C    INPUT: N - SAMPLE SIZE
C           X,Y - IN COMMON BLOCK
C           PSIQ - INTEGRATING FACTOR FOR BHAT
C           LOCO - LOCATION OF FIRST COEFFICIENT IN COF FOR BEST
C                  MODEL
C           KPARM - NUMBER OF PARAMETERS IN BEST MODEL
C           COF,IND - VECTOR OF COEFFICIENTS AND INDICES FOR BEST
C                     MODEL
C
C    OUTPUT: RQ - VECTOR CONTAINING REGRESSION QUANTILE FUNCTION
C
C    AUXILLIARY: E
C
C    SUBPROGRAMS CALLED: PPLOT,QUENT,DESTAT,MAX,MIN,MINMAX,QTOFQ,
C                        QFIND,QUICK,WSPACE,FQFNC
C
C*****************************************************************
      COMMON /DATAR/ X(500),Y(500),RANKX(500),RANKY(500)
      DIMENSION IND(1),RQ(1),E(1)
      DIMENSION T(500,2),U(500),QY(500)
      DIMENSION LABR(20),LABRQ(20),
     +UNAME(20),YNAME(20),YNAMX(20),XNAME(20),CHAR(5)
      COMPLEX CEXP,CMPLX,ZARG,COF(97)
      DATA NTDIM/500/
      DATA CHAR/'*','+','.','X','O'/
      DATA LABR/'REGR','ESSI','ON F','UNCT','ION,',' * -',' OBS',
     +'ERVE','D, +',' - P','REDI','CTED',8*'    '/
      DATA LABRQ/'REGR','ESSI','ON Q','UANT','ILE ','FUNC','TION',
     +', * ',' - OB','SERV','ED, ','+ - ','PRED','ICTE','D   ',6*'    '/
      DATA UNAME/8*' ','U','1',10*' '/
      DATA XNAME/8*' ','X',11*' '/
      DATA YNAME/7*' ','R','Q','-','Y',9*' '/
      DATA YNAMX/8*' ','R','-','Y',8*' '/
      REAL LSDHAT
      TWOPI=8.0*ATAN(1.0)
C
C    ORDER Y AS RAW ESTIMATE OF QUANTILE FUNCTION
C
      DO 10 I=1,N
   10 QY(I)=Y(I)
      CALL QUICK(N,QY)
      DIVN=1./FLOAT(N)
      DO 40 I=1,N
      U1=(FLOAT(I)-.5)*DIVN
      RQ(I)=0.
      DO 30 J=1,N
      U2=(FLOAT(J)-.5)*DIVN
      LSDHAT=0.0
      LOC=LOCO
```

```
      DO 15 K=1,KPARM
      I1=IND(LOC)-1
      I2=MOD(I1,M)
      I1=(I1-I2)/M-ML
      I2=I2-ML
      ARG=TWOPI*(FLOAT(I1)*U1+FLOAT(I2)*U2)
      ZARG=CMPLX(0.0,ARG)
      LGDHAT=LGDHAT+REAL(CBF(LOC)*CEXP(ZARG))
      LOC=LOC+1
   15 CONTINUE
      IF(LGDHAT.LE.170.) GO TO 25
      WRITE(6,20) U1,U2,LGDHAT
   20 FORMAT(/,10X,'LOG DHAT(',F8.6,',',F8.6,')=',E20.6,
     *' IN QREG. QUANTILE REGRESSION TERMINATED.')
      RETURN
   25 IF(LGDHAT.LT.-20.) LGDHAT=-20.
      DHAT=EXP(LGDHAT)/PSIQ
      RO(I)=RO(I)+QV(J)*DHAT
   30 CONTINUE
      RO(I)=RO(I)*DIVN
   40 CONTINUE
      U(1)=(FLOAT(1)-0.5)*DIVN
      T(1,1)=V(1)
      T(1,2)=RO(1)
      DO 60 I=2,N
      U(I)=U(I-1)+DIVN
      T(I,1)=V(I)
      T(I,2)=RO(I)
   60 CONTINUE
      CALL PPLOT(U,T,NTDIM,N,2,CHAR,LABRO,UNAME,YNAME,1)
      CALL PPLOT(X,T,NTDIM,N,2,CHAR,LABR,XNAME,YNAMX,1)
      SMSQ1=0.
      DO 70 I=1,N
      E(I)=V(I)-RO(I)
   70 SMSQ1=SMSQ1+E(I)*E(I)
      SMSQ1=SMSQ1/FLOAT(N-1)
      WRITE(6,100) SMSQ1
  100 FORMAT(/,10X,'RESIDUAL VARIANCE = ',F15.6)
      RETURN
      END
      SUBROUTINE QTOFQ(Q,U,N,XS,SPCFAC)
C**********************************************************
C   SUBROUTINE TO COMPUTE LITTLE Q(U) FROM
C   THE EMPIRICAL QUANTILE FN CAP Q(U) AND THE U VALUES.
C   INPUT : Q,U,NQ
C   OUTPUT : XS : SPACINGS=LITTLE Q(U)
C************************************************************
      DIMENSION Q(N),U(N),XS(N)
      S = 1.E10
      EPS = 1.E-8
      DU = 1. / (U(N) - U(N-1))
      XS(1) = (Q(2) - Q(1)) / (U(2) - U(1))
      IF(XS(1) .GT. EPS) S = XS(1)
      XS(N-1) = (Q(N) - Q(N-1)) * DU
      IF(XS(N-1) .LT. S .AND. XS(N-1) .GT. EPS) S = XS(N-1)
      DU = DU * .5
      NM2 = N-2
      DO 10 I = 2,NM2
      XS(I) = (Q(I+2) - Q(I)) * DU
      IF(XS(I) .LT. S .AND. XS(I) .GT. EPS) S = XS(I)
10    CONTINUE
      NM1 = N-1
      S = S * SPCFAC
      DO 20 I = 1,NM1
      IF(XS(I) .LE. EPS) XS(I) = S
20    CONTINUE
      RETURN
      END
      SUBROUTINE QUENT(N,X,NAME,LAB,INUL,NDIM,X25,X50,X75,XBAR,SD)
C*****************************************************************
C
C   SUBROUTINE TO COMPUTE QUANTILE AND ENTROPY STATISTICS FOR
C   DATA SET X.
C
C   INPUT: X,N - DATA SET X OF LENGTH N
C          NAME - NAME FOR ANALYSIS IN 20A4 FORMAT
C          LAB - LABEL FOR DATA SET IN 20A4 FORMAT
C          INUL - NULL DISTRIBUTIONS FOR X (SEE FOFNC)
C          NDIM - DIMENSION OF X IN CALLING PROGRAM
C
C   OUTPUT: X25,X50,X75 - LOWER QUARTILE, MEDIAN, AND UPPER QUARTILE
C           XBAR,SD - SAMPLE MEAN AND STANDARD DEVIATION
C
C   SUBPROGRAMS CALLED: QUICK,DESTAT,PPLOT,FOFNC,WSPACE,KSD,FCODEA
C
C*****************************************************************
      COMMON /PARM/ BETAP,BETAW
      DIMENSION NAME(20),IFORM(20),LICASE(22)
      DIMENSION X(NDIM),Y(552),QN(552),U(552),FQ(552),D(552),
     &          WK1(552),WK2(552),WK3(552)
      DIMENSION LAB(20),LAB1(20),LAB5(20),LAB9(20),LAB10(20)
      DIMENSION NAMU(2),NAM10(2),NAMCWS(2),ICASE(2)
      DIMENSION LQUART(4),EX10(8),EXU(8),STRT(2),END(2),CHARL(2)
      DATA EXU / 0.01,0.05,0.10,0.25,0.75,0.90,0.95,0.99 /
      DATA LICASE/4HNORM,4HAL  ,4HEXPO,4HNTAL,4HLOGI,4HSTIC,
     &            4HDBL ,4HEXP ,4HUN R,4HECIP,4HCAUC,4HHY  ,
     &            4HEX V,4HALUE,4HLOG ,4HNORM,4HPARE,4HTO  ,
     &            4HWEIB,4HULL ,4HHALF,4H LOG/
      DATA BLK/' '/
      DATA STRT,END,CHARL /-.5,0.,.5,0.,'*','-'/
      DATA ASTER/1H*/
      DATA LAB1/'INFO','RMAT','IVE ','QUAN','TILE',' - ',14*'    '/
      DATA LAB5/4HD* =,2*4H    ,4H AT ,4HU = ,2*4H    ,4H.  D,
     *4H- = ,2*4H    ,4H AT ,4HU = ,7*4H    /
      DATA LAB10/4HRAW ,4H DIS,4HTRIB,4HUTIO,4HN D(,4HU)  ,
     *     3*4H    ,4H CAS,4HE   ,9*4H    /
      DATA NAMU,NAM10/ 4H    ,4H   U,4H  10,4H(U) /
      DATA NAMCWS/4H    ,4HD(U)/
      DATA SPCFAC/0.1/
      NUNIT = 6
      DO 5 I=1,12
    5 LAB1(I+8)=LAB(I)
      WRITE(NUNIT,1000)
 1000 FORMAT(///,T50,'FULLY  NON-PARAMETRIC  ANALYSIS'/T50,31(1H*))
C   OBTAIN DESCRIPTIVE STATISTICS FOR UNGROUPED DATA
      XONE = 1.E+10
      PI = 4.*ATAN(1.)
      TWOPI = PI + PI
      RPI = 1./PI
      RSQTPI = 1./SQRT(TWOPI)
C
      NQ=N-1
      NQP1=N
      DO 10 I=1,N
   10 Y(I)=X(I)
      CALL QUICK(N,Y)
      YMIN=Y(1)
      CALL DESTAT(Y,N,NAME,LAB,LQUART,0,1,X25,X50,X75)
      FN=1./FLOAT(N)
      U(1)=.5*FN
      DO 50 I=1,N
      U(I+1)=U(I)+FN
```

```
   20 ON(I)=Y(I)
      T10=1./(2.*(X75-X25))
      DO 30 I=1,N
      ON(I)=(ON(I)-X50)*T10
      WK1(I)=U(I)-.5
   30 WK2(I)=ON(I)
      DO 40 I=1,N
      IF(WK2(I).GE.-1.) GOTO 50
   40 WK2(I)=-1.
   50 DO 60 I=1,N
      IF(WK2(N-I+1).LE.1.) GOTO 70
   60 WK2(N-I+1)=1.
   70 CONTINUE
      CALL OPLOT(U,WK2,NOP1,BLK,0.,0.,1,0,BLK,2,-1.,1.,2,STRT,END,
     &           CHARL,NAMU,NAMIO,NAME,LAB1)
      DO 75 I = 1,8
      EXIO(I) = OFIND(ON,NOP1,EXU(I))
   75 CONTINUE
      WRITE(NUNIT,80) EXU,EXIO
   80 FORMAT(//T10,'  U  ',3X,8(F8.5,1X),
     *        /T10,'IO(U)',3X,8(F8.5,1X),///)
      KASE=INUL*2
      ICASE(1)=LICASE(KASE-1)
      ICASE(2)=LICASE(KASE)
C   COMPUTE AND PLOT RAW SPACINGS (=O) AND RAW FO (=1./O)
      CALL OTOFO(ON,U,NOP1,WK1,SPCFAC)
C   COMPUTE AND PLOT WEIGHTED SPACINGS FOR CASE ICASE
      DO 100 I = 1,NO
      WK2(I) = FOPNC(U(I+1),INUL)
  100 CONTINUE
      CALL WSPACE(WKS,D,NOP1,WK1,WK2,U,SW)
C   PLOT CUMULATIVE WEIGHTED SPACINGS WITH D+ AND D-
      CALL KSD(D,U,NOP1,DM,UM,DP,UP)
      CALL FCODEA(DP,8H(F7.4)  ,LAB9(2))
      CALL FCODEA(UP,8H(F7.4)  ,LAB9(6))
      CALL FCODEA(DM,8H(F7.4)  ,LAB9(10))
      CALL FCODEA(UM,8H(F7.4)  ,LAB9(14))
      LAB10(8)=ICASE(1)
      LAB10(9 )=ICASE(2)
      CALL OPLOT(U,D,NOP1,BLK,0.,0.,1,0,BLK,2,0.,1.,1,0.,1.,ASTER,
     &           NAMU,NAMCWS,LAB10,LAB9)
      WRITE(NUNIT,130)
  130 FORMAT(//////)
      RETURN
      END
      SUBROUTINE QUICK(N,T)
C*************************************************************************
C   QUICK SORT.THIS ALGORITH IS ALSO REFERRED TO AS A PARTITIONED
C   EXCHANGE SORT.EXPECTED RUNTIME IS PROPORTIONAL TO N*LOG2(N)
C   ALTHOUGH THE WORST CASE IS PROPORTIONAL TO N**2.
C   REFERENCE: DONALD E. KNUTH- THE ART OF COMPUTER PROGRAMMING VOL 3.
C   INPUT :
C          X,N : VECTOR TO BE SORTED OF LENGTH N
C   OUTPUT :
C          X : SORTED VECTOR
C   SUBROUTINES CALLED : NONE
C*************************************************************************
      REAL T(N),Y
      INTEGER IP,LV(16),IV(16),LP,IUP
      LV(1)=1
      IV(1)=N
      IP=1
   10 IF(IP.LT.1) GO TO 75
   15 IF((IV(IP)-LV(IP)).LT.1) GO TO 20
      GO TO 25
   20 IP=IP-1
      GO TO 10
   25 LP=LV(IP)-1
      IUP=IV(IP)
      Y=T(IUP)
   30 IF((IUP-LP).LT.2) GO TO 45
      LP=LP+1
      IF(T(LP).LE.Y) GO TO 30
      T(IUP)=T(LP)
   35 IF((IUP-LP).LT.2) GO TO 40
      IUP=IUP-1
      IF(T(IUP).GE.Y) GO TO 35
      T(LP)=T(IUP)
      GO TO 30
   40 IUP=IUP-1
   45 T(IUP)=Y
      IF((IUP-LV(IP)).LT.(IV(IP)-IUP)) GO TO 55
      GO TO 60
   55 LV(IP+1)=LV(IP)
      IV(IP+1)=IUP-1
      LV(IP)=IUP+1
      GO TO 70
   60 LV(IP+1)=IUP+1
      IV(IP+1)=IV(IP)
      IV(IP)=IUP-1
   70 IP=IP+1
      GO TO 15
   75 RETURN
      END
      SUBROUTINE RANK(X,N,XR)
C
C*******************************************************************
C
C   SUBROUTINE TO RANK THE N-VECTOR X WITH RANKS PLACED IN
C   THE N-VECTOR XR.
C
C   TIED VALUES ARE GIVEN AVERAGE RANKS.
C
C   SUBROUTINE CALLED: ORD2
C
C*******************************************************************
C
      DIMENSION X(N),XR(N),W1(500,2),W2(500,2),Y(500)
C
C   CREATE MATRIX W1 WHOSE FIRST COLUMN CONTAINS THE VECTOR X
C   AND WHOSE SECOND COLUMN CONTAINS THE OBSERVATION NUMBER
C
      DO 10 I=1,N
      W1(I,1)=X(I)
      W1(I,2)=FLOAT(I)
   10 CONTINUE
      CALL ORD2(W1,N,500)
C
C   CREATE MATRIX W2 WHOSE FIRST COLUMN CONTAINS SORTED X VALUES
C   AND WHOSE SECOND COLUMN CONTAINS THE RANKS OF THE X VALUES
C   BEFORE CORRECTING FOR TIES.
C
      DO 20 I=1,N
      W2(I,1)=W1(I,1)
      W2(I,2)=FLOAT(I)
   20 CONTINUE
C
C   CORRECT FOR TIES BY REPLACING RANKS OF TIED VALUES BY
C   AVERAGE RANK
C
      I=1
   30 K=I+1
```

```
   40 IF(K.GT.N) GO TO 50                                                DISAM
C                                                                        DISAM
C   REAL ARITHMETIC TOLERANCE FACTOR: ASSUME X=0 IF ABS(X).LE.5.E-6     DISAM
C                                                                        DISAM
      WTEST=ABS(W2(I,1)-W2(K,1))                                         DISAM
      IF(WTEST.GT.5.E-6) GO TO 50                                        DISAM
      K=K+1                                                              DISAM
      GO TO 40                                                           DISAM
   50 K2=K-1                                                             DISAM
      K3=K-I                                                             DISAM
      IF(K3.LE.1) GO TO 80                                               DISAM
      SUM=0.                                                             DISAM
      DO 60 J=I,K2                                                       DISAM
      SUM=SUM+W2(J,2)                                                    DISAM
   60 CONTINUE                                                           DISAM
      DO 70 J=I,K2                                                       DISAM
      V(J)=SUM/FLOAT(K3)                                                 DISAM
   70 CONTINUE                                                           DISAM
      GO TO 90                                                           DISAM
   80 V(I)=W2(I,2)                                                       DISAM
   90 I=K                                                                DISAM
      IF(I.LT.N) GO TO 30                                                DISAM
      WTEST2=ABS(W2(N,1)-W2(N-1,1))                                      DISAM
      IF(WTEST2.LT.5.E-6) GO TO 100                                      DISAM
      V(N)=FLOAT(N)                                                      DISAM
C                                                                        DISAM
C   CREATE VECTOR XR CONTAINING RANKS CORRECTED FOR TIES                 DISAM
C                                                                        DISAM
  100 DO 110 I=1,N                                                       DISAM
      KW=W1(I,2)+0.5                                                     DISAM
      XR(KW)=V(I)                                                        DISAM
  110 CONTINUE                                                           DISAM
      RETURN                                                             DISAM
      END                                                                DISAM
      SUBROUTINE SPRMN(N,RHO,SUMD)                                       DISAM
C                                                                        DISAM
C*************************************************************           DISAM
C                                                                        DISAM
C   SUBROUTINE TO COMPUTE SPEARMAN'S RHO.                                DISAM
C                                                                        DISAM
C   INPUT: RANKX - THE VECTOR OF RANKS OF THE INDEPENDENT VARIABLE      DISAM
C          RANKY - THE VECTOR OF RANKS OF THE DEPENDENT VARIABLE        DISAM
C          N - THE NUMBER OF PAIRED OBSERVATIONS                         DISAM
C                                                                        DISAM
C   OUTPUT: RHO - SPEARMAN'S RANK CORRELATION COEFFICIENT.              DISAM
C                                                                        DISAM
C*************************************************************           DISAM
C                                                                        DISAM
      COMMON /DATAR/ X(500),Y(500),RANKX(500),RANKY(500)                 DISAM
C                                                                        DISAM
C   COMPUTE SQUARE OF RANK DIFFERENCES                                   DISAM
C                                                                        DISAM
      SUMD=0.                                                            DISAM
      DO 10 I=1,N                                                        DISAM
      DIF=RANKX(I)-RANKY(I)                                              DISAM
      SUMD=SUMD+DIF*DIF                                                  DISAM
   10 CONTINUE                                                           DISAM
C                                                                        DISAM
C   COMPUTE SPEARMAN'S RANK CORRELATION COEFFICIENT                      DISAM
C                                                                        DISAM
      SNUM=6.*SUMD                                                       DISAM
      SDENOM=FLOAT(N)*(FLOAT(N*N)-1.)                                    DISAM
      RHO=1.-SNUM/SDENOM                                                 DISAM
      RETURN                                                             DISAM
      END                                                                DISAM
      SUBROUTINE TRIM(X,Y,XMED,YMED,KDEL,N,NEWN)                         DISAM
C*********************************************************               DISAM
C                                                                        DISAM
C   SUBPROGRAM TO TRIM A BIVARIATE DATA SET OF AT MOST                   DISAM
C   KDEL "EXTREME" POINTS BASED ON DISTANCE FROM THR                     DISAM
C   MEDIAN IN THE X AND Y DIRECTIONS ONLY.                               DISAM
C                                                                        DISAM
C   INPUT: X,Y - DATA OF SIZE N                                          DISAM
C          XMED,YMED - MEDIANS OF X AND Y                                DISAM
C          KDEL - MAXIMUM NUMBER OF POINTS TO DELETE FROM DATA SET       DISAM
C                                                                        DISAM
C   OUTPUT: X,Y - TRIMMED DATA OF SIZE NEWN                              DISAM
C                                                                        DISAM
C   SUBPROGRAMS CALLED: MAX,MIN,QUICK,ORD2                               DISAM
C                                                                        DISAM
C*********************************************************               DISAM
      DIMENSION X(N),Y(N),DELM(20,2),XDEL(20),IDEL(20),YY(500)           DISAM
      NEWN=N                                                             DISAM
      IF(KDEL.LE.0) RETURN                                               DISAM
      KCHK=0                                                             DISAM
      IF(KDEL.NE.1) GO TO 3                                              DISAM
      KCHK=1                                                             DISAM
      KDEL=2                                                             DISAM
    3 IF(KDEL.LT.N) GO TO 6                                              DISAM
      WRITE(6,5)                                                         DISAM
    5 FORMAT(10X,'KDEL IS GREATER THAN OR EQUAL TO N'./,                 DISAM
     +10X,'KDEL HAS BEEN SET EQUAL TO 4.')                               DISAM
      KDEL=4                                                             DISAM
    6 IF(KDEL.GT.20) KDEL=20                                             DISAM
      KMOD=MOD(KDEL,2)                                                   DISAM
      IF(KMOD.EQ.1) KDEL=KDEL-1                                          DISAM
      KDIM=KDEL                                                          DISAM
      KHALF=KDEL/2                                                       DISAM
      DO 10 I=1,KHALF                                                    DISAM
      J=N-I+1                                                            DISAM
      DELM(I,1)=XMED-X(I)                                                DISAM
      DELM(I,2)=FLOAT(I)                                                 DISAM
      DELM(KDEL-I+1,1)=X(J)-XMED                                         DISAM
      DELM(KDEL-I+1,2)=FLOAT(J)                                          DISAM
   10 CONTINUE                                                           DISAM
      CALL ORD2(DELM,KDEL,20)                                            DISAM
      DO 20 I=1,KHALF                                                    DISAM
      IDEL(I)=IFIX(DELM(KHALF+I,2)+0.5)                                  DISAM
   20 CONTINUE                                                           DISAM
      IF(KCHK.EQ.1) GO TO 65                                             DISAM
      DO 30 I=1,N                                                        DISAM
      YY(I)=Y(I)                                                         DISAM
   30 CONTINUE                                                           DISAM
      CALL MIN(Y,N,YMIN,IMIN)                                            DISAM
      CALL MAX(Y,N,YMAX,IMAX)                                            DISAM
      DO 40 I=1,KHALF                                                    DISAM
      DELM(I,1)=YMED-YMIN                                                DISAM
      DELM(I,2)=FLOAT(IMIN)                                              DISAM
      Y(IMIN)=YMAX                                                       DISAM
      CALL MIN(Y,N,YMIN,IMIN)                                            DISAM
   40 CONTINUE                                                           DISAM
      DO 50 I=1,N                                                        DISAM
      Y(I)=YY(I)                                                         DISAM
   50 CONTINUE                                                           DISAM
      CALL MIN(Y,N,YMIN,IMIN)                                            DISAM
      DO 60 I=1,KHALF                                                    DISAM
      DELM(KDEL-I+1,1)=YMAX-YMED                                         DISAM
      DELM(KDEL-I+1,2)=FLOAT(IMAX)                                       DISAM
      Y(IMAX)=YMIN                                                       DISAM
      CALL MAX(Y,N,YMAX,IMAX)                                            DISAM
   60 CONTINUE                                                           DISAM
      DO 65 I=1,N                                                        DISAM
```

```
   65 Y(I)=YY(I)                                                        BISAM
      CALL ORD2(DELM,KDEL,20)                                           BISAM
      DO 70 I=1,KHALF                                                   BISAM
      IDEL(KHALF+I)=IFIX(DELM(KHALF+I,2)+0.5)                           BISAM
   70 CONTINUE                                                          BISAM
      KH1=KHALF+1                                                       BISAM
      DO 80 I=1,KHALF                                                   BISAM
      DO 80 J=KH1,KDIM                                                  BISAM
      IF(IDEL(I).NE.IDEL(J)) GO TO 80                                   BISAM
      IDEL(J)=N+1                                                       BISAM
      KDEL=KDEL-1                                                       BISAM
   80 CONTINUE                                                          BISAM
      DO 82 I=1,KDIM                                                    BISAM
   82 XDEL(I)=FLOAT(IDEL(I))                                            BISAM
      CALL QUICK(KDIM,XDEL)                                             BISAM
      DO 84 I=1,KDIM                                                    BISAM
   84 IDEL(I)=IFIX(XDEL(I)+0.5)                                         BISAM
   85 CONTINUE                                                          BISAM
      IF(KCHK.EQ.1) KDEL=1                                              BISAM
      NN1=N-1                                                           BISAM
      DO 100 I=1,KDEL                                                   BISAM
      IND=IDEL(KDEL-I+1)                                                BISAM
      NEWN=NEWN-1                                                       BISAM
      IF(IND.EQ.N) GO TO 92                                             BISAM
      DO 90 J=IND,NN1                                                   BISAM
      X(J)=X(J+1)                                                       BISAM
      Y(J)=Y(J+1)                                                       BISAM
   90 CONTINUE                                                          BISAM
   92 X(NEWN+1)=0.0                                                     BISAM
      Y(NEWN+1)=0.0                                                     BISAM
      NN1=NEWN                                                          BISAM
  100 CONTINUE                                                          BISAM
      WRITE(6,110)                                                      BISAM
  110 FORMAT(10X,'THE FOLLOWING POINTS WERE DELETED FROM THE DATA SET:')BISAM
      WRITE(6,120) (IDEL(I),I=1,KDEL)                                   BISAM
  120 FORMAT(10X,8I5)                                                   BISAM
      WRITE(6,130) KDEL,NEWN                                            BISAM
  130 FORMAT(/,10X,I5,' POINTS WERE DELETED LEAVING ',I5,' POINTS ',    BISAM
     +'IN THE DATA SET.')                                               BISAM
      RETURN                                                            BISAM
      END                                                               BISAM
      SUBROUTINE WSPACE(WXS,CWXS,NOP1,XS,FOOO,U,SIGO,AVLWX)             BISAM
C*************************************************************          BISAM
C   SUBROUTINE TO COMPUTE D(U), CUMULATIVE D'S, AND SIGMAO              BISAM
C   FOR THE MODEL Q(U)=MU+SIGMA*Q0(U)                                   BISAM
C   INPUT :                                                             BISAM
C             XS,NO : VECTOR OF LENGTH NO CONTAINING LITTLE Q(U)        BISAM
C             FOOO : HYPOTHESIZED DENSITY QUANTILE FUNCTION             BISAM
C             U : VECTOR OF LENGTH NO CONTAINING U VALUES               BISAM
C   OUTPUT :                                                            BISAM
C             WXS : VECTOR OF LENGTH NO CONTAINING D(U)                 BISAM
C             CWXS : VECTOR OF LENGTH NO CONTAINING THE                 BISAM
C                    CUMULATIVE D'S                                     BISAM
C             SIGO : COMPUTED VALUE OF SIGMAO = CWXS(NO)                BISAM
C                                                                       BISAM
C   SUBROUTINES CALLED : NONE                                           BISAM
C*************************************************************          BISAM
      DIMENSION FOOO(NOP1),XS(NOP1),U(NOP1),WXS(NOP1),CWXS(NOP1)        BISAM
      NO=NOP1-1                                                         BISAM
      CWXS(1)=0.                                                        BISAM
      DO 10 I= 1,NO                                                     BISAM
      WXS(I) = FOOO(I) * XS(I)                                          BISAM
      CWXS(I+1) = CWXS(I) + WXS(I)                                      BISAM
   10 CONTINUE                                                          BISAM
      FN = FLOAT(NO)                                                    BISAM
      D1 = 1./CWXS(NOP1)                                                BISAM
      D2 = D1 * FN                                                      BISAM
      DO 20 I=1,NO                                                      BISAM
      WXS(I)=WXS(I) * D2                                                BISAM
      CWXS(I)=CWXS(I) * D1                                              BISAM
   20 CONTINUE                                                          BISAM
      CWXS(NOP1)=1.                                                     BISAM
      SIGO = 1./D2                                                      BISAM
      RETURN                                                            BISAM
      END                                                               BISAM
```

REFERENCES

Kimeldorf, George, and Sampson, Allen R. (1975), "Uniform Representations of Bivariate Distributions," Comm. Stat., 4, 617-628.

Newton, H. Joseph (1979), "The TIMESBOARD Time Series Analysis Computing Library," Institute of Statistics, Texas A&M University, College Station, Texas.

Parzen, Emanuel (1979), "Nonparametric Statistical Data Modeling," JASA, 74, 105-131.

Parzen, Emanuel, and Anderson, Scott (1980), "ONESAM, A Computer Program for Nonparametric Data Analysis and Density Estimation," Technical Report B-1, Institute of Statistics, Texas A&M University, College Station, Texas.

Prihoda, Thomas J. (1981), "A Generalized Approach to the Two Sample Problem: The Quantile Approach," unpublished Ph.D. dissertation, Institute of Statistics, Texas A&M University.

SAS/GRAPH User's Guide (1981), SAS Institute Inc., Cary, North Carolina.

Tartar, M. E., and Kronmal, R. A. (1970), "On Multivariate Density Estimates Based on Orthogonal Expansions," Annals of Math. Stat., 41, 718-722.

_______________ (1976), "An Introduction ot the Implementation and Theory of Nonparametric Density Estimation," American Statistician, 30, 105-112.

Woodfield, Terry J. (1982), "Statistical Modeling of Bivariate Data," unpublished Ph.D. dissertation, Institute of Statistics, Texas A&M University.